

**T.C.
MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**DESTEK VEKTÖR REGRESYON İLE
UÇAKLARDA ANLIK YAKIT TÜKETİMİNİN ANALİZİ**

DOKTORA TEZİ


Cansın YALÇIN

İstatistik Anabilim Dalı

Tez Danışmanı: Prof. Dr. Gülay BAŞARIR

Haziran 2016

Cansın YALÇIN tarafından hazırlanan DESTEK VEKTÖR REGRESYON İLE UÇAKLARDA ANLIK YAKIT TÜKETİMİNİN ANALİZİ adlı bu tezin Doktora tezi olarak uygun olduğunu onaylarım.


Prof. Dr. Gülay BAŞARIR
Tez Yöneticisi

Bu çalışma, jürimiz tarafından İstatistik Anabilim Dalında Doktora tezi olarak kabul edilmiştir.

Başkan: : Prof. Dr. Gülay BAŞARIR

Üye : Prof. Dr. Aydın ERAR

Üye : Prof. Dr. Müjgan TEZ

Üye : Prof. Dr. Ahmet Mete ÇİLİNGİRTÜRK

Üye : Doç. Dr. Eylem DENİZ HOWE

Bu tez, Mimar Sinan Güzel Sanatlar Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygundur.

DESTEK VEKTÖR REGRESYON İLE UÇAKLARDA ANLIK YAKIT TÜKETİMİNİN ANALİZİ

ÖZET

Makine öğrenme algoritmaları son zamanlarda özellikle on binlerce değişken ve/veya gözlemin mevcut olduğu veri setleri üzerinde çalışma yapan araştırmacıların ilgi odağı olmuştur. Destek vektör makineleri maksimum marjin prensibine dayanan çekirdek-tabanlı makine öğrenme algoritmalarıdır. Temelini istatistiksel öğrenme teorisinden almakta ve deneysel risk minimizasyonu yerine yapısal risk minimizasyonu prensibini kullanmaktadır. Bu özelliği sayesinde yerel minimuma takılma ve aşırı uyum problemleri aşılmıştır.

Destek vektör makineleri ilk olarak sınıflandırma problemi için önerilmiş olup temel olarak doğrusal ikili sınıflandırma yöntemidir. Diskriminant fonksiyonu konveks optimizasyon probleminin çözülmesi ile bulunmaktadır. Destek vektör makineleri daha sonra regresyon ve anomali tespiti problemleri için geliştirilmiş ve büyük boyutlu verilerde başarılı sonuçlar verdiği gösterilmiştir.

Nonlinear ilişkilerin tespitinde veri daha yüksek boyutlu özellik uzayına haritalanmaktadır. Dual formulasyonda gözlemlerin sadece iç çarpımlar şeklinde yer alması özelliğinden faydalanılmakta ve problemin çözümünde çekirdek fonksiyonları kullanılmaktadır. Destek vektörler ile temsil edilen modelin kompleksliği girdi örneklerinin boyutundan bağımsızdır ve sadece destek vektörlerin sayısına bağlı olmaktadır.

Bu çalışmada, destek vektör regresyon yönteminin uçuş süresince anlık yakıt tüketiminin tahmini üzerinde uygulaması yapılmıştır. Yakıt tüketiminin Uçuş Veri İzleme sistemlerinde kaydedilen verilerin nonlinear bir fonksiyonu olarak

modellenebileceği gösterilmiştir. Uygulanan model gerçek veri seti üzerinde test edilmiştir. Regresyon problemlerinin çözümünde kullanılan karar ağaçları, yapay sinir ağları ve Gaussian süreç yöntemlerinin aynı veri seti üzerinde uygulamaları sunulmuş ve modellerin birbirlerine olan üstünlükleri incelenmiştir.

Sonuç olarak, makine öğrenme regresyon yöntemlerinin uçaklarda anlık yakıt tüketiminin tahmini probleminin çözümünde başarılı bir şekilde uygulanabileceği gösterilmiştir. Destek vektör regresyon ile kurulan model en yüksek performans gösteren sonuçları vermiştir. Yakıt tüketimine ilişkin açıklayıcı değişkenlerin her saniye kaydedildiği ve parametre sayısının fazlalığı göz önünde bulundurulduğunda destek vektör makineleri yönteminin seyrek (sparse) sonuçlar vermesi ve modeli verinin boyutundan bağımsız olarak sadece destek vektör olarak seçilen gözlemlere dayanarak oluşturması maliyet etkin çözümler sunmaktadır.

Nonlineer çekirdek tabanlı destek vektör makinelerinin kullanılması uçuşun tüm safhalarının istatistiksel analizini mümkün kılmıştır. Bu yöntem mevcut durumda uygulanan sadece toplam yakıt tüketim miktarının planlanan toplam yakıt tüketim miktarı ile karşılaştırılmasından ziyade uçuş süresince tüketilen yakıt miktarının modellenmesi ile gerçekleşen ve beklenen anlık yakıt tüketim miktarlarının analizine dayanmaktadır. Elde edilen regresyon modelinden faydalanılarak, gerçekleşen yakıt tüketiminin beklenen değerin üzerinde olduğu durumlar ve aşırı yakıt tüketimine neden olan faktörler tespit edilebilmektedir. Uygulanan yöntem incelenmek istenen nicel bir değişkenin ve bu değişken üzerinde etkisi olan kategorik veya nicel değişkenlerin zaman serisi ölçümlerinin mevcut olduğu tüm sistemler için kullanılabilir.

ANALYSIS OF INSTANTANEOUS FUEL CONSUMPTION IN AIRCRAFTS WITH SUPPORT VECTOR REGRESSION

SUMMARY

Machine learning algorithms have recently been the focus of researchers especially those working on data sets with tens of thousands of variables and/or observations. Support Vector Machines (SVMs) algorithm is a kernel-based machine learning method based on maximum margin principle. It is developed in the framework of statistical learning theory using structural risk minimization principle instead of empirical risk minimization. Thus, SVMs can overcome the overfitting and local minima problems.

SVMs are first proposed for classification problems and basically are linear binary classifiers. The decision boundary can be found by solving a convex optimization problem. Later, SVMs for regression and anomaly detection problems are developed and proved to give high accuracy results on large data sets.

To detect nonlinear relationship, the data is mapped to a higher dimensional feature space. Dual formulation can be expressed in terms of inner products of data instances and so the problem can be solved by using kernel functions. The complexity of the model represented by support vectors is independent from the size of input space and depends only on the number of support vectors.

In this study, support vector regression algorithm is applied to predict the instantaneous fuel consumption during flight. It has been shown that fuel consumption can be modeled as a nonlinear function of data collected in Flight Data Monitoring systems. The model has been tested on real-world data set. Other regression methods,

such as decision trees, artificial neural networks and Gaussian process, are also applied on the same data set for comparison purposes.

To conclude, machine learning regression methods are proved to be successful to predict instantaneous fuel consumption in aircrafts. The Support Vector Regression model yielded the results with the highest accuracy. Taken into consideration the large number of parameters and that the explanatory variables are saved in every second during a flight, SVMs present efficient solutions by providing a sparse model which is independent from the size of the input space and depends only on the number of support vectors.

Using nonlinear kernel functions in Support Vector Regression enabled statistical analysis of all phases of the flight. This method is based on modeling fuel consumption during flight and comparison of the expected and consumed instantaneous fuel amount, instead of the current implementation which merely compares the total fuel consumption with the planned total amount. Using Support Vector Regression model, the over consumption as well as the causing factors can be detected. The model can be applied on any system which has time series measurements of continuous response variable and measurements of related categorical and/or quantitative variables.

ÖNSÖZ

Lisansüstü eğitimime başladığım ilk günden itibaren hem alanındaki bilgi ve tecrübesi hem de takdire şayan kişiliği ile bana örnek olan Tez Danışmanım ve aynı zamanda Mimar Sinan Güzel Sanatlar Üniversitesi İstatistik Bölüm Başkanı Prof. Dr. Gülay BAŞARIR'a verdiği tüm desteklerden dolayı teşekkürü borç bilirim.

Doktora eğitimime başlamam için beni cesaretlendiren ve üzerimde çok emeği olan Değerli Hocam Fen Edebiyat Fakültesi Dekanı Prof. Dr. Nalân CİNEMRE'ye, Doktora Tez Jürimde yer alan Prof. Dr. Aydın ERAR, Prof. Dr. Müjgân TEZ, Prof. Dr. Ahmet Mete ÇİLİNGİRTÜRK ve Doç. Dr. Eylem DENİZ HOWE'ye vermiş oldukları faydalı geri bildirimler, değerli yorumları ve yönlendirmeleri için içten teşekkürlerimi sunarım.

Lisansüstü eğitimime başlamama vesile olan, hayata karşı duruşunu hayranlıkla izlediğim ve tüm başarılarımı borçlu olduğum Canım Annem Gülseren DURLU'ya ve enerjisiyle zor anlarımızda bile bizi gülümsetmeyi başaran Canım Kardeşim Dr. Sinem Sultan KAYA'ya her zaman yanımda oldukları ve her konuda benden desteklerini esirgemedikleri için sonsuz teşekkür ederim.

Doktora eğitimim boyunca Türkiye'de ve Amerika Birleşik Devletleri'nde, hayatımın her anında yanımda olan, zor zamanlarımda beni hiçbir zaman yalnız bırakmayan, bana her zaman inanan ve cesaretlendiren, hayatıma girdiği için kendimi çok şanslı hissettiğim Canım Eşim Mustafa YALÇIN'a desteklerinden ve gösterdiği anlayıştan dolayı sonsuz teşekkür ederim.

Bu Doktora Tezi, Fulbright Doktora Tez Araştırma Bursu ile desteklenmiştir. Çok emek verilerek hazırlanmış olan bu çalışmanın Destek Vektör Makineleri ve Uçuş Veri İzleme sistemleri alanlarında çalışan araştırmacılara faydalı bir kaynak olmasını dilerim.

Haziran, 2016

Cansın YALÇIN

İÇİNDEKİLER

ÖZET	i
SUMMARY	iii
ÖNSÖZ	v
ÇİZELGE LİSTESİ	viii
ŞEKİL LİSTESİ	ix
SEMBOL LİSTESİ	xi
KISALTMALAR LİSTESİ	xii
1. GİRİŞ	1
2. LİTERATÜR ARAŞTIRMASI	6
3. DESTEK VEKTÖR MAKİNELERİ	9
3.1. Destek Vektör Makineleri - İkili Sınıflandırma	9
3.1.1. Doğrusal Destek Vektör Makineleri	9
3.1.2. Esnek (Yumuşak) Marjin Destek Vektör Makineleri	11
3.1.3. Dual Problem	11
3.1.4. Nonlineer Destek Vektör Makineleri	13
3.2. Destek Vektör Regresyon	15
3.2.1. Doğrusal Destek Vektör Regresyon	15
3.2.2. Esnek (Yumuşak) Marjin Destek Vektör Regresyon	16
3.2.3. Dual Problem	17
3.2.4. Nonlineer Destek Vektör Regresyon	19
3.2.5. Kayıp Fonksiyonları	20
3.2.6. Duyarlılık Parametresinin Seçimi	21
3.2.7. Otomatik Hata Kontrolü	22
3.3. Tek Sınıf Destek Vektör Makineleri	24
3.4. Çoklu Grup Sınıflandırma Destek Vektör Makineleri	25
3.5. Çekirdek Yöntemi	26
3.5.1. Yüksek Boyutlu Uzaya Haritalama	27
3.5.2. Çekirdek Matrisi	28
3.5.3. Mercer Koşulları	28
3.5.4. Çekirdek Fonksiyonları	29
3.6. Optimizasyon Algoritmaları	29
3.6.1. İç Noktalar Algoritmaları	29
3.6.2. Alt Küme Seçme Algoritmaları	30
3.6.3. Sıralı Minimal Optimizasyon	30

4. ÇOKLU ÇEKİRDEK ÖĞRENME ALGORİTMALARI	32
4.1. Çoklu Çekirdek Algoritmalarının Sınıflandırılması	33
4.2. Destek Vektör Regresyon için Uygulanan Çoklu Çekirdek Algoritmaları	36
4.2.1. Çekirdek Fonksiyonlarının Kombinasyonu için Sabit Kurallar	38
4.2.2. Negatif Olmayan Doğrusal Kombinasyon (Konik Kombinasyon)	38
4.2.3. Konveks Kombinasyon	45
4.2.4. Çekirdek Fonksiyonlarının Nonlinear Kombinasyonu	51
4.2.5. Veriye Bağlı Çoklu Çekirdek Algoritmaları	53
5. DEĞİŞKEN SEÇİMİ	56
5.1. Değişken Derecelendirme	57
5.2. Değişken Alt Küme Seçimi	58
5.2.1. Sarmalama ve Gömülü Yöntemler	58
5.2.2. İç İçe Geçmiş Alt Kümeler Yöntemi	59
5.2.3. Doğrudan Amaç Optimizasyonu	62
5.2.4. Alt Küme Seçimi için Filtreleme Yöntemleri	63
5.3. Destek Vektör Regresyon için Değişken Seçimi	63
6. UYGULAMA	66
6.1. Havacılıkta Yakıt Tüketimi	67
6.1.1. Havacılıkta Yakıt Tasarrufu: Çevresel ve Ekonomik Bakış Açısı	67
6.1.2. Uçuş Veri İzleme Sistemleri	68
6.1.3. Uçuş Veri Setinin Tanıtılması	71
6.1.4. Uçuşun Safhaları	74
6.2. Verinin Hazırlanması	74
6.2.1. Veri Ölçeklendirme	77
6.3. Tek Çekirdek Fonksiyonu Kullanılarak Elde Edilen Sonuçlar	78
6.3.1. Aşırı Yakıt Tüketimine Sebep Olan Faktörlerin Araştırılması	84
6.4. Çoklu Çekirdek Fonksiyonu Kullanılarak Elde Edilen Sonuçlar	89
6.4.1. Gaussian Çekirdek Fonksiyonlarının Konik Kombinasyonu	89
6.4.2. Polinomial Çekirdek Fonksiyonlarının Konik Kombinasyonu	92
6.4.3. Farklı Çekirdek Fonksiyonlarının Konveks Kombinasyonu	94
6.5. Diğer Regresyon Yöntemleri ile Karşılaştırma	95
6.5.1. Regresyon Ağaçları	95
6.5.2. Lasso Düzleştirme	102
6.5.3. Yapay Sinir Ağları	107
6.5.4. Gaussian Süreç Regresyon	111
7. SONUÇ VE ÖNERİLER	115
KAYNAKLAR	119
ÖZGEÇMİŞ	127

ÇİZELGE LİSTESİ

	Sayfa No
Çizelge 4.1. Destek vektör makineleri için uygulanan çoklu çekirdek algoritmaları	37
Çizelge 6.1. Uçuş süre ve mesafeleri.....	75
Çizelge 6.2. Modelde kullanılan açıklayıcı değişkenler	75
Çizelge 6.3. Hata tüpünün genişliği = 0,1 ile elde edilen sonuçlar	79
Çizelge 6.4. Hata tüpünün genişliği = 0,01 ile elde edilen sonuçlar	80
Çizelge 6.5. Gaussian çekirdek fonksiyonu için farklı ceza katsayıları ile elde edilen sonuçlar	81
Çizelge 6.6. Polinomial çekirdek fonksiyonu için farklı ceza katsayıları ile elde edilen sonuçlar	82
Çizelge 6.7. Gaussian çekirdek fonksiyonu için farklı genişlik değerleri ile elde edilen sonuçlar	89
Çizelge 6.8. Polinomial çekirdek fonksiyonu için farklı parametre değerleri ile elde edilen sonuçlar	92
Çizelge 6.9. Regresyon ağacı modeli ile elde edilen sonuçlar	98
Çizelge 6.10. Temel model ve optimum model sonuçlarının karşılaştırılması	98
Çizelge 6.11. Regresyon ağacı modeli için açıklayıcı değişkenlerin önem dereceleri	100
Çizelge 6.12. Lasso ve doğrusal regresyon model sonuçlarının karşılaştırılması.....	106
Çizelge 7.1. Regresyon model sonuçlarının karşılaştırılması.....	117

ŞEKİL LİSTESİ

Sayfa No

Şekil 1.1. Kümeleme problemi	2
Şekil 1.2. Çekirdek yöntemi	3
Şekil 3.1. Doğrusal ayırıcı hiperdüzlem	10
Şekil 3.2. Özellik uzayına haritalama.....	13
Şekil 3.3. İkili sınıflandırma destek vektör makineleri	15
Şekil 3.4. Doğrusal destek vektör regresyon için esnek marjin kayıp fonksiyonu	17
Şekil 3.5. Gaussian çekirdek fonksiyonu ile destek vektör regresyon	20
Şekil 3.6. Kayıp fonksiyonları.....	21
Şekil 3.7. Duyarlılık parametresinin regresyon kestirimine etkisi	22
Şekil 3.8. Duyarlılık parametresinin destek vektör sayısı üzerindeki etkisi.....	22
Şekil 3.9. Çoklu grup sınıflandırma destek vektör makineleri	26
Şekil 3.10. Yüksek boyutlu uzaya haritalama	27
Şekil 3.11. Çekirdek matrisi	28
Şekil 4.1. Sinüs dalgası problemi	48
Şekil 4.2. Sinüs fonksiyonlarının kombinasyonu	49
Şekil 4.3. Yerel çoklu çekirdek algoritmaları.....	55
Şekil 6.1. Uçuş veri izleme süreci	70
Şekil 6.2. Hücüm açısı ve yunuslama açısı	72
Şekil 6.3. Flap konfigürasyonu.....	73
Şekil 6.4. Uçuşun safhaları	74
Şekil 6.5. İç hatlar uçuşları için artıklar	83
Şekil 6.6. Dış hatlar uçuşları için NHKOK değerleri	83
Şekil 6.7. Aşırı yakıt tüketimi yapılan uçuşlara ait yakıt tüketimi grafiği	85
Şekil 6.8. Normal yakıt tüketimi yapılan uçuşa ait yakıt tüketimi grafiği	86
Şekil 6.9. Destek vektörler	88
Şekil 6.10. Gaussian çekirdek fonksiyonu için farklı genişlik değerleri ile elde edilen sonuçlar	90

Şekil 6.11. Farklı genişlikte Gaussian çekirdek fonksiyonlarına atanan ağırlıklar	91
Şekil 6.12. Polinomial çekirdek fonksiyonu için farklı parametre değerleri ile elde edilen sonuçlar	93
Şekil 6.13. Farklı dereceli Polinomial çekirdek fonksiyonlarına atanan ağırlıklar	94
Şekil 6.14. Gaussian ve Polinomial çekirdek fonksiyonlarına atanan ağırlıklar	94
Şekil 6.15. Örnek regresyon ağacı modeli.....	95
Şekil 6.16. Farklı minimum yaprak büyüklükleri için çapraz geçerlilik hata değerleri.....	98
Şekil 6.17. Açıklayıcı değişkenlerin önem dereceleri	101
Şekil 6.18. Regresyon ağaçları toplulukları ve hata değerleri.....	102
Şekil 6.19. Çapraz geçerlilik grafiği.....	104
Şekil 6.20. Lasso iz grafiği.....	105
Şekil 6.21. Doğrusal regresyon modeli elde edilen sonuçlar	106
Şekil 6.22. Yapay sinir ağı modelinin eğitilmesi	107
Şekil 6.23. Yapay sinir ağı modeli	108
Şekil 6.24. Regresyon grafikleri.....	109
Şekil 6.25. Hataların histogram grafiği	110
Şekil 6.26. Yapay sinir ağları modeli ile elde edilen sonuçlar	110
Şekil 6.27. Gaussian süreç regresyon uyumu.....	113
Şekil 6.28. Gaussian süreç regresyon modeli ile elde edilen sonuçlar	114

SEMBOL LİSTESİ

x	: Veri örneği
y	: Çıktı değişkeni
N	: Örnek sayısı
\mathbb{R}	: Reel sayılar
D	: Girdi uzayının boyutu
w	: Ağırlık katsayıları
b	: Yan terim
H	: Hiperdüzlem
C	: Ceza katsayısı
ξ	: Gevşek değişken
ε	: Hata tüpünün genişliği
α, β	: Lagrange çarpanları
$\Phi(\cdot)$: Haritalama fonksiyonu
$\langle \cdot, \cdot \rangle$: İç çarpım
$k(\cdot, \cdot)$: Çekirdek fonksiyonu
\mathbf{K}	: Çekirdek matrisi
p	: Polinomial çekirdek fonksiyonunun derecesi
σ	: Gaussian çekirdek fonksiyonunun genişliği
P	: Birleştirilen çekirdek fonksiyonu sayısı
η	: Çekirdek fonksiyonunun ağırlığı
\top	: Transpoz
\odot	: Hadamard çarpımı

KISALTMALAR

DP	: Doğrusal Programlama
DV	: Destek Vektör
DVR	: Destek Vektör Regresyon
HKO	: Hata Kareler Ortalaması
İDKP	: İkinci Dereceden Konik Programlama
KKKP	: Karesel olarak Kısıtlı Karesel Programlama
KKT	: Karush-Kuhn-Tucker
KP	: Karesel Programlama
NHKOK	: Normalleştirilmiş Hata Kareler Ortalamasının Karekökü
UVİ	: Uçuş Veri İzleme
YBP	: Yarı-Belirli Programlama
YSDP	: Yarı-Sonsuz Doğrusal Programlama

1. GİRİŞ

Makine öğrenme (machine learning), bilgisayarlara açıkça programlanmış olmadan öğrenme yeteneği veren çalışma alanıdır (Samuel, 1959). Tom Mitchell (1997) makine öğrenme için şu tanımı kullanmıştır: Eğer bir bilgisayar programının, G sınıfında bulunan görevlerdeki (P ile ölçülen) performansı, T tecrübe ile artıyor ise bu bilgisayar programı G görevler sınıfı ve P performans ölçüsüne göre T tecrübeden öğreniyor demektir. Örneğin bir e-posta programının kullanıcının hangi e-postaları spam (istenmeyen) olarak işaretlediğini/işaretlemediğini takip ettiğini ve buna dayanarak istenmeyen e-postaları nasıl daha iyi filtreleyeceğini öğrendiğini varsayalım. Bu makine öğrenme probleminde “görev” e-postaların spam veya spam değil olarak sınıflandırılmasıdır. E-postaların spam/spam değil olarak etiketlenmesinin izlenmesi “tecrübe”, doğru sınıflandırılan e-postaların sayısı (veya oranı) ise “performans ölçüsü”dür.

Makine öğrenme yöntemleri verinin yapısına göre ikiye ayrılır: danışmanlı öğrenme (supervised learning) ve danışmansız öğrenme (unsupervised learning). Danışmanlı öğrenmede makine öğrenme algoritmasına eğitim için girdi verisi ve gerçek çıktı (etiket, hedef) değerleri verilmektedir ve amaç girdiler ile çıktılar arasındaki genel ilişkinin öğrenilmesidir. Danışmansız öğrenmede ise makine öğrenme algoritmasına çıktı bilgisi verilmemektedir ve amaç girdi verisi içindeki yapının (örüntünün) öğrenilmesidir.

Makine öğrenme problemleri için bir diğer kategorizasyon da model çıktısına göre yapılmaktadır. Sınıflandırma problemlerinde girdi örnekleri iki veya daha fazla sınıfa ayrılmakta ve elde edilen model yeni örneklerin hangi sınıfa veya çoklu etiket sınıflandırma (multi-label classification) problemlerinde hangi sınıflara ait olduğuna karar vermek için kullanılmaktadır. Sınıflandırma problemleri genellikle danışmanlı makine öğrenme problemleridir. Örneğin spam filtreleme amacıyla kullanılan e-posta programı bir sınıflandırma problemidir. Sınıflandırma problemlerinin çözümünde kullanılan makine öğrenme yöntemlerinden bazıları şunlardır: destek vektör

makineleri, yapay sinir ağları, karar ağaçları, k-en yakın komşu, lojistik regresyon ve diskriminant analizi.

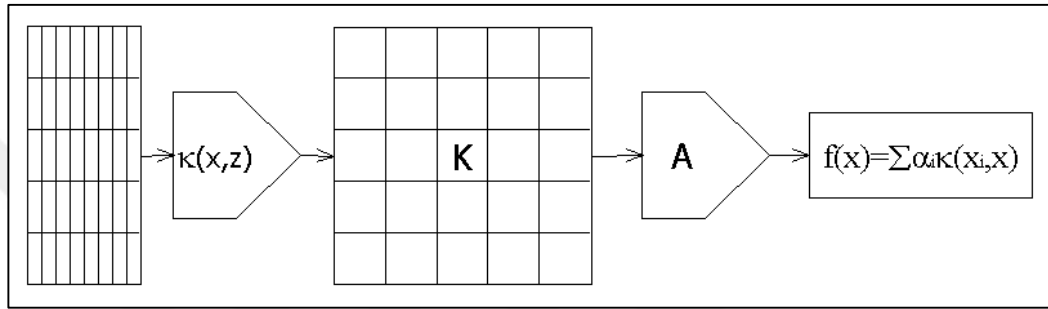
Regresyon problemleri de danışmanlı makine öğrenme problemidir, ancak burada model çıktısı nicel bir değişkendir. Örneğin bu tezde uygulaması ele alınacak olan uçuş veri izleme sistemlerinde kaydedilen ve uçağın durumunu gösteren değişken değerleri kullanılarak uçağın anlık yakıt tüketiminin tahmin edilmesi problemi bir regresyon problemidir. Regresyon problemlerinin çözümünde kullanılan makine öğrenme yöntemlerinden bazıları şunlardır: destek vektör regresyon, yapay sinir ağları, karar ağaçları ve Gaussian süreç (Gaussian process).

Kümeleme problemlerinde ise girdi örnekleri gruplara bölünmektedir. Sınıflandırma problemlerinden farklı olarak burada sınıflar önceden bilinmemektedir. Dolayısıyla kümeleme problemleri danışmansız makine öğrenme problemleridir. Kümeleme problemi için ilgi çekici bir örnek Google'ın haber sitesi (news.google.com.tr)'dir. Google her gün farklı kaynaklarda yayınlanan binlerce haberi aramakta ve aynı konu ile ilgili olanları bir grupta toplamaktadır. Şekil 1.1 incelenecek olursa iPad Pro ile ilgili başlık altında farklı bağlantılar bulunduğu görülmektedir. Bu bağlantılar okuyucuyu farklı kaynaklarda aynı konu, bu spesifik durum için iPad Pro, ile ilgili yayınlanan haberlere ulaştıracaktır. İnternette bulunan haber makalelerinin aynı olayı anlatan makaleler bir küme oluşturacak şekilde otomatik olarak gruplandırılması bir kümeleme problemidir. Kümeleme problemlerinin çözümünde kullanılan makine öğrenme yöntemlerine örnek olarak k-ortalama yöntemi verilebilir.



Şekil 1.1. Kümeleme problemi

Çekirdek yöntemleri örüntü tanımlama (pattern recognition) problemlerinin çözümünde kullanılan algoritmalarıdır. En önemli özelliklerinden birisi genel veri türleri (dizi, vektör, metin, resim vb.) için uygulanabilmesi ve genel ilişki tiplerinin (sınıflandırma, regresyon, kümeleme vb.) tespitinde kullanılabilmesidir. Çekirdek yöntemlerinin uygulanması Şekil 1.2’de gösterilen iki temel aşamadan oluşmaktadır. İlk adımda veri setinden çekirdek fonksiyonu yardımıyla çekirdek matrisi oluşturulmakta, ikinci adımda ise sadece çekirdek matrisinde mevcut olan bilgi kullanılarak veri analizi için kullanılacak olan çekirdek algoritması uygulanmaktadır.



Şekil 1.2. Çekirdek yöntemi (Cristianini ve Shawe-Taylor, 2004)

Çekirdek fonksiyonu örnekler arasındaki iç çarpımları hesaplamaktadır. Yani doğrusal algoritmadaki iç çarpımlar çekirdek fonksiyonu ile değiştirilebilir. Böylece hem yöntemin nonlinear (doğrusal olmayan) ilişkilerin tespitinde ve vektör olmayan veri tipleri üzerinde kullanılması sağlanmakta hem de hesaplama etkinliği kazanılmaktadır. Çekirdek algoritmalarının uygulandığı yöntemlerden bazıları şunlardır: destek vektör makineleri, ridge regresyon, diskriminant analizi, temel bileşenler analizi ve kanonik korelasyon analizi.

Destek vektör makineleri maksimum marjın prensibine dayanan çekirdek-tabanlı makine öğrenme algoritmalarıdır. Temelini istatistiksel öğrenme teorisinden almakta ve deneysel risk minimizasyonu yerine yapısal risk minimizasyonu prensibini kullanmaktadır. Bu özelliği sayesinde yerel minimuma takılma ve aşırı uyum problemleri aşılmıştır. Destek vektör makineleri ilk olarak sınıflandırma problemi için önerilmiş olup temel olarak doğrusal ikili sınıflandırma yöntemidir. Diskriminant fonksiyonu konveks optimizasyon probleminin çözülmesi ile bulunmaktadır. Nonlinear ilişkilerin tespitinde veri nonlinear bir fonksiyon kullanılarak daha yüksek boyutlu özellik uzayına haritalanmaktadır. Dual formülasyonda gözlemlerin sadece iç

çarpımlar şeklinde yer alması özelliğinden faydalanılmakta ve problemin çözümünde benzerlik ölçüsü olarak çekirdek fonksiyonları kullanılmaktadır.

Destek vektör makineleri daha sonra regresyon ve anomali tespiti problemleri için geliştirilmiş ve büyük boyutlu verilerde başarılı sonuçlar verdiği gösterilmiştir. Destek vektör makinelerinde en büyük problem sahası model seçimi, yani çekirdek fonksiyonunun ve optimum parametre değerlerinin seçimidir. Bu problemin aşılması için çoklu çekirdek öğrenme algoritmaları önerilmiştir. Çoklu çekirdek öğrenme algoritmalarında kullanıcı tarafından tek bir çekirdek fonksiyonu belirlenmesi yerine bir çekirdek fonksiyonu seti belirlenmekte ve model bu çekirdek fonksiyonlarının bir kombinasyonunu kullanmaktadır.

Bu çalışmada destek vektör regresyon yönteminin uçaklarda anlık yakıt tüketiminin tahmini üzerinde uygulaması yapılmıştır. Yakıt tüketiminin Uçuş Veri İzleme (Flight Data Monitoring - FDM) sistemlerinde kaydedilen verilerin nonlineer bir fonksiyonu olarak modellenebileceği gösterilmiştir.

Uygulanan yöntem incelenmek istenen nicel bir değişkenin ve bu değişken üzerinde etkisi olan kategorik veya nicel değişkenlerin zaman serisi ölçümlerinin mevcut olduğu tüm sistemler için kullanılabilir. Bu yöntem mevcut durumda uygulanan sadece toplam yakıt tüketim miktarının planlanan toplam yakıt tüketim miktarı ile karşılaştırılmasından ziyade uçuş süresince tüketilen yakıt miktarının modellenmesi ile gerçekleşen ve beklenen anlık yakıt tüketim miktarlarının analizine dayanmaktadır.

Yapılan araştırmalar dâhilinde bu tez Uçuş Veri İzleme sistemlerinde kaydedilen veriyi kullanarak uçuş süresince anlık yakıt tüketimini istatistiksel olarak modelleyen Türkiye'deki ilk çalışmadır. Uçaklarda anlık yakıt tüketiminin tahmin edilmesinde ilk defa nonlineer çekirdek tabanlı destek vektör makineleri kullanılmış ve ilk defa uçuşun tüm safhaları istatistiksel olarak analiz edilmiştir. Böylece kullanılan yöntem uçuşun her safhasının değerlendirilebilmesine olanak sağlamaktadır. Uygulanan model gerçek veri seti üzerinde test edilmiştir.

Bu çalışmada amaç uçuş süresince anlık yakıt tüketiminin analizi için destek vektör regresyon ve çekirdek yönteminin uygulanmasıdır. Bu amaç doğrultusunda izleyen 2'nci Bölümde destek vektör makineleri, çoklu çekirdek öğrenme algoritmaları ve uçaklarda yakıt tüketimi alanlarında yapılmış olan çalışmalar detaylı bir şekilde incelenmiştir. 3'üncü Bölümde destek vektör makineleri yönteminin sınıflandırma,

regresyon ve anomali tespiti problemlerinin çözümünde uygulanması açıklanmış, doğrusal olmayan ilişkilerin analizi için çekirdek yöntemi modele dâhil edilmiştir. 4'üncü Bölümde çekirdek fonksiyonu seçimi problemine çözüm olarak çoklu çekirdek algoritmaları ele alınmıştır. 5'inci Bölümde değişken seçimi problemi için kullanılan yöntemlere yer verilmiştir. 6'ıncı Bölümde yöntemin uçak yakıt tüketimi verisi üzerinde uygulaması sunulmuştur. Son bölümde ise elde edilen sonuçlar analiz edilmiş ve üzerinde çalışılması planlanan konular tartışılmıştır.



2. LİTERATÜR ARAŞTIRMASI

Destek vektör makineleri, sınıflandırma ve örüntü tanımlama problemleri için el yazısı rakam tanımlamada (Cortes ve Vapnik, 1995; Schölkopf ve diğ., 1995 ve 1996; Burges ve Schölkopf, 1997), nesne tanımlamada (Blanz ve diğ., 1996), konuşmacı tanımlamada (Schmidt, 1996), görüntülerde yüz tespitinde (Osuna ve diğ., 1997a) ve metin kategorizasyonunda (Joachims, 1997) kullanılmıştır. Örüntü tanımlama için destek vektör makineleri üzerine kapsamlı bir çalışma Burges (1998) tarafından yapılmıştır.

Destek vektör regresyon Boston ev problemi (Drucker ve diğ., 1997) ve zaman serileri (Müller ve diğ., 1997; Mukherjee ve diğ., 1997) problemlerinin çözümünde kullanılmıştır. Destek vektör makinelerinin regresyon problemlerinin çözümünde kullanılmasına ilişkin kapsamlı bir çalışma Smola ve Schölkopf (2004) tarafından yapılmıştır.

Çekirdek yöntemi için Cristianini ve Shawe-Taylor (2000 ve 2004) ile Schölkopf ve Smola (2002) tarafından yazılan kitaplar temel oluşturmakta, hem teori hem de uygulama ile ilgili detaylı kaynak sağlamaktadır.

Gönen ve Alpaydın (2011) çoklu çekirdek öğrenme algoritmalarını öğrenme yöntemi, fonksiyonel form, hedef fonksiyonu, eğitim yöntemi, temel öğrenme metodu ve hesaplama kompleksliği özellikleri bakımından sınıflandırmışlardır. Gönen (2010) tarafından destek vektör makineleri için yerel çoklu çekirdek algoritmaları önerilmiştir.

Karal (2011) tarafından gürbüz ve yalın regresyon için parçalı doğrusal ve destek vektör tabanlı modeller kullanılmıştır. Kabaoğlu (2010) tarafından destek vektör makineleri tabanlı hata bulma, tanıma ve hata toleranslı kontrol yöntemleri önerilmiştir. Ayhan (2013) tarafından nitelik indirgeme ve sınıflandırma problemlerinin çözümü için kaba küme ve destek vektör makineleri yaklaşımı uygulanmıştır.

Çoklu çekirdek algoritmaları destek vektör makinelerinde en büyük problem sahası olan model seçimi, yani çekirdek fonksiyonu ve ilgili parametrelerinin seçimi, konusunda etkin çözümler sağlanmakta ve elde edilen sonuçların yorumlanmasına olanak tanımaktadır. Ancak yapılan araştırmalar kapsamında Türkiye’de çoklu çekirdek algoritmaları üzerine yapılan çalışmaların çok nadir olduğu görülmüştür. Bu nedenle tezin ileri aşamalarında destek vektör makineleri için çoklu çekirdek algoritmaları konusu detaylı olarak ele alınacaktır.

Bununla birlikte önerilen çoklu çekirdek algoritmaları genellikle sınıflandırma problemlerine yöneliktir. Tezin uygulama aşamasında yöntemin yakıt tüketimi probleminin çözümünde kullanılması ele alınarak çoklu çekirdek algoritmalarının destek vektör regresyon yöntemi için uygulanışı da çalışılmış olacaktır.

NASA (The National Aeronautics and Space Administration - Ulusal Havacılık ve Uzay Dairesi)’nin devam eden yeşil havacılık projesi kapsamında Srivastava (2012) uçak yakıt tüketiminin tahmini ve anomali (anomaly) tespiti için topluluk (ensemble) öğrenme regresyon modellerini kullanmıştır. Uçak yakıt tüketimini Uçuş Operasyonel Kalite Güvencesi (Flight Operation Quality Assurance - FOQA) sistemlerinde kaydedilen verilerin bir fonksiyonu olarak tanımlamıştır. Gaussian süreç regresyon, sinir ağları ve elastik ağlar (elastic nets) yöntemlerini kullanmıştır. Güven aralığının belirlenmesinde hataların Gaussian dağılıma uyduğu varsayılmıştır. Uçuşun kalkış ve iniş safhalarında hava trafik kontrolünün etkisini ortadan kaldırmak ve tüm uçuşların aynı uzunlukta olmasını sağlamak amacıyla sadece uçuşun iniş takımları kapandıktan sonraki ilk 25 dakikasını incelemiştir.

Woodbury ve Srivastava (2012) tarafından yapılan çalışmada uçak yakıt tüketiminin tahmini ve anomali tespiti için kullanılan üç algoritma şunlardır: genelleştirilmiş lineer model, yapay sinir ağları ve Gaussian süreç. Bootstrap örnekleme metodunu uygulamışlardır. Regresyon modelleri topluluğundaki tüm modellerden elde edilen tahminlerin ortalamasını hesaplamışlardır. Tahmin için güven aralıklarını topluluk tahmininin standart sapmasının 3 katı olarak almışlardır. Bu aralığın üzerindeki yakıt tüketimlerini anomali olarak değerlendirmişlerdir. Model doğruluğunu değerlendirmek için normalleştirilmiş hata kareler ortalamasının karekökü değerini kullanmışlardır. Uygulanan yöntemlerin normalleştirilmiş hata kareler ortalamasının karekökü değerlerinin %20’nin altında olduğunu bulmuşlardır.

Das ve diğ. (2014) uçak yakıt tüketiminin tahmini ve anomali tespiti için Parito-SVR (Parallel Iterated Optimizer for Support Vector Regression in the Primal) yöntemini kullanmışlardır. Yakıt tüketiminin tahmini için Uçuş Operasyonel Kalite Güvencesi sistemlerinde kaydedilen parametreleri kullanmışlardır. Yanıt değişkeni olarak iki motordaki toplam yakıt akış miktarını almışlardır. Uçuşun sadece düz uçuş (cruise) aşamasını modellemişlerdir. Destek vektör regresyon yöntemini kullanmışlar ancak primal problemi çözmüşlerdir. Yani veriyi nonlinear bir fonksiyon ile daha yüksek boyutlu bir uzaya haritalamamışlar, diğer bir deyişle çekirdek fonksiyonu kullanmamışlardır. Düz uçuş aşamasındaki yakıt tüketim grafiği incelendiğinde doğrusal destek vektör regresyonun yeterli olduğu anlaşılmaktadır. Sonraki çalışmalarda daha iyi modelleme sağlayarak modelin performansını geliştireceği değerlendirildiğinden nonlinear çekirdek tabanlı destek vektör makinelerinin kullanılmasını önermişlerdir.

Anomali tespiti için en büyük problem veri setinde anormal olarak etiketlenmiş uçuşların bulunmamasıdır. Oza ve Das (2014) eğitim setindeki anormal uçuşları elemek için şöyle bir yöntem izlemiştir: Modeli eğitim verisi üzerinde test etmişler, uçuşun her saniyesi için hata kareler ortalamasının karekökünü hesaplamışlar ve bu değerlerin ortalamasını alarak o uçuş için ortalama hata kareler ortalamasının karekökünü hesaplamışlardır. Bu işlemi bütün uçuşlar için tekrarlamışlardır. Bunun sonucunda bütün uçuşlar için ortalama hata kareler ortalamasının karekökünü bulmuşlar ve bunun için 3 sigma sınırını hesaplamışlardır. Ortalama hata kareler ortalamasının karekökü bu 3 sigma sınırının üzerinde olan uçuşları eğitim setinden çıkarmışlardır. Test verisindeki anormal uçuşları tespit etmek için de aynı yöntemi kullanmışlardır.

3. DESTEK VEKTÖR MAKİNELERİ

Destek vektör makineleri (Support vector machines) Boser ve diğ. (1992) tarafından önerilmiştir. İstatistiksel öğrenme teorisi (statistical learning theory) ve yapısal risk minimizasyonu prensibine dayanır (Vapnik, 1995 ve 1998).

Bu bölümde sırası ile ikili sınıflandırma, regresyon, tek sınıf ve çoklu grup sınıflandırma problemlerinin çözümünde destek vektör makinelerinin uygulanması açıklanacaktır.

3.1. DESTEK VEKTÖR MAKİNELERİ - İKİLİ SINIFLANDIRMA

Bu kısımda öncelikle doğrusal olarak ayrılabilen verilerin sınıflandırılması ve doğrusal olarak ayrılamayan durumda esnek (yumuşak) marjin destek vektör makinelerinin uygulanması açıklanacaktır. Daha sonra problemin dual formülasyonu sunulacak ve nonlineer destek vektör makineleri incelenecektir.¹

3.1.1. Doğrusal Destek Vektör Makineleri

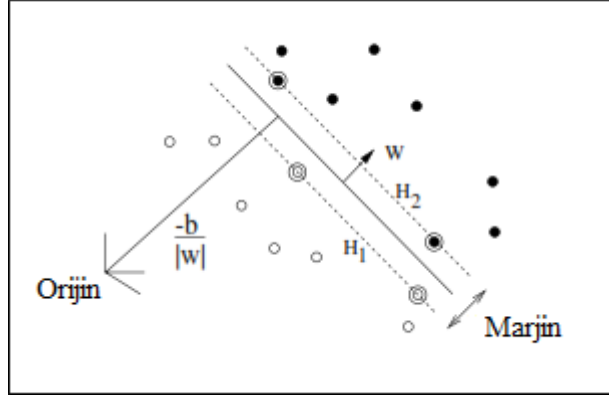
Örüntü tanımlama için destek vektör makineleri üzerine kapsamlı bir çalışma Burges (1998) tarafından yapılmıştır.

İlk olarak sınıfların doğrusal olarak ayrılabilir olduğu durum ele alınsın. Bağımsız olarak çekilmiş ve aynı dağılıma sahip eğitim örnekleri $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ şeklinde verilmiş olsun. Burada $\mathbf{x}_i \in \mathbb{R}^D$, D boyutlu açıklayıcı değişkenler ve $y_i \in \{+1, -1\}$ sınıf etiketleridir. Pozitif ve negatif sınıf etiketlerine sahip eğitim örneklerini ayıran bir ayırıcı hiperdüzlem olduğu varsayalım. Diskriminant (ayırma) fonksiyonu aşağıdaki şekilde yazılır:

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b \quad (3.1)$$

¹ Koyu renk yazı tipi vektör ve matrisleri, normal yazı tipi vektör, matris elemanlarını ve sayısal değerleri ifade etmek için kullanılmıştır.

Burada $\mathbf{w} \in \mathbb{R}^D$ ağırlıklar vektörü, $b \in \mathbb{R}$ ise diskriminant fonksiyonunun yan terimidir. İki boyutlu uzayda çözüm Şekil 3.1’de gösterildiği şekilde olacaktır.



Şekil 3.1. Doğrusal ayırıcı hiperdüzlem (Burges, 1998)

Ayırıcı hiperdüzlem üzerindeki \mathbf{x} örnekleri $\mathbf{w} \cdot \mathbf{x} + b = 0$ kısıtını sağlamaktadır. Burada \mathbf{w} hiperdüzlemin normalidir. $|b|/\|\mathbf{w}\|$ hiperdüzlemden orijine dik uzaklıktır. $\|\mathbf{w}\|$, \mathbf{w} 'nin öklid normudur. d_+ (d_-) ayırıcı hiperdüzlemden en yakın pozitif (negatif) örneğe en kısa uzaklık olmak üzere ayırıcı hiperdüzlemin marjini ($d_+ + d_-$) olarak tanımlanır. Doğrusal olarak ayrılabilir durumda destek vektör makineleri maksimum marjine sahip ayırıcı hiperdüzlemi aramaktadır. Tüm eğitim örneklerinin aşağıdaki kısıtları sağladığı varsayalım.

$$\mathbf{w} \cdot \mathbf{x}_i + b \geq +1 \quad y_i = +1 \text{ için} \quad (3.2)$$

$$\mathbf{w} \cdot \mathbf{x}_i + b \leq -1 \quad y_i = -1 \text{ için} \quad (3.3)$$

Bu iki kısıt aşağıdaki şekilde birleştirilebilir:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \quad \forall i \quad (3.4)$$

Eşitlik (3.2) ile verilen kısıt eşitlik şeklinde sağlayan örnekler $H_1: \mathbf{x}_i \cdot \mathbf{w} + b = +1$ hiperdüzlemi üzerinde yer alırlar ve orijinden dik uzaklıkları $|+1 - b|/\|\mathbf{w}\|$ olur. Benzer şekilde Eşitlik (3.3) ile verilen kısıt eşitlik şeklinde sağlayan örnekler $H_2: \mathbf{x}_i \cdot \mathbf{w} + b = -1$ hiperdüzlemi üzerinde yer alırlar ve orijinden dik uzaklıkları $|-1 - b|/\|\mathbf{w}\|$ olur. Dolayısıyla $d_+ = d_- = 1/\|\mathbf{w}\|$ ve marjin $2/\|\mathbf{w}\|$ 'dir. Burada H_1 ve H_2 paraleldir, aynı normale sahiptirler ve bu iki hiperdüzlem arasında hiç bir eğitim örneği yer almamaktadır. Böylece $\|\mathbf{w}\|^2$ ifadesi Eşitlik (3.4) ile verilen kısıtları sağlayacak şekilde minimize edilerek maksimum marjine sahip hiperdüzlem

bulunabilir. Eşitlik (3.1) ile verilen diskriminant fonksiyonunun parametreleri aşağıdaki karesel optimizasyon probleminin çözülmesi ile bulunur:

$$\begin{aligned} \min \quad & \frac{1}{2} \|\mathbf{w}\|^2 \\ & y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \quad \forall i \\ & \mathbf{x}_i \in \mathbb{R}^D, y_i \in \{+1, -1\}, \mathbf{w} \in \mathbb{R}^D, b \in \mathbb{R} \end{aligned} \quad (3.5)$$

Şekil 3.1’de görüldüğü üzere çözüm sadece Eşitlik (3.4) ile verilen kısıtı eşitlik şeklinde sağlayan, H_1 veya H_2 düzlemi üzerinde bulunan, eğitim örneklerine bağlıdır. Bu örnekler destek vektörler olarak adlandırılırlar (Burges, 1998).

3.1.2. Esnek (Yumuşak) Marjın Destek Vektör Makineleri

Doğrusal olarak ayırlamayan verinin söz konusu olduğu durumda çözümün bulunabilmesi için Eşitlik (3.2) ve (3.3) ile verilen kısıtlara ξ_i pozitif gevşek değişkenleri eklenerek kısıtlar gevşetilir ve buna bağlı olarak amaç fonksiyonuna C ceza katsayısı ilave edilir (Cortes ve Vapnik, 1995). Elde edilen model esnek (yumuşak) marjın destek vektör makineleri olarak adlandırılmaktadır.

Karesel optimizasyon problemi aşağıdaki şekilde yazılır:

$$\begin{aligned} \min \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \\ & y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \xi_i \geq 0 \quad \forall i \end{aligned} \quad (3.6)$$

3.1.3. Dual Problem

Primal problemin Lagrange dual formu aşağıdaki şekilde elde edilir (Fletcher, 2000):

$$L = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i (y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 + \xi_i) - \sum_{i=1}^N \beta_i \xi_i \quad (3.7)$$

Burada $\alpha_i, \beta_i \geq 0$ Lagrange çarpanlarıdır. L 'nin primal parametrelere göre türevi alınırsa aşağıdaki eşitlikler elde edilir:

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i \quad (3.8)$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^N \alpha_i y_i = 0 \quad (3.9)$$

$$\frac{\partial L}{\partial \xi_i} = 0 \Rightarrow C = \alpha_i + \beta_i \quad \forall i \quad (3.10)$$

Eşitlik (3.7) ve (3.8) ile verilen ifadelerden faydalanılarak dual model aşağıdaki şekilde elde edilir:

$$\begin{aligned} & maks \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\ & \sum_{i=1}^N \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq C \quad \forall i \end{aligned} \quad (3.11)$$

Yukarıdaki optimizasyon probleminin çözülmesi ile ağırlıklar bulunur.

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i \quad (3.12)$$

\mathbf{x} test örneği olmak üzere diskriminant fonksiyonu aşağıdaki şekilde yazılır:

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \quad (3.13)$$

Burada önemli bir nokta problemin Lagrange formülasyonunda eğitim verilerinin sadece iç çarpımlar şeklinde yer almasıdır. Bu özellik modelin nonlineer duruma genelleştirilmesinde kullanılacaktır.

Her eğitim örneği için bir α_i Lagrange çarpanı olduğu görülmektedir. Çözümde sadece $\alpha_i > 0$ olan, yani H_1 veya H_2 düzlemi üzerinde yer alan ve destek vektörler olarak adlandırılan örnekler yer alacaktır. Destek vektörler, destek vektör makineleri için kritik elemanlardır. Diğer örneklerin veri setinden çıkarılması, H_1 veya H_2

hiperdüzlemlerini kesmeyecek şekilde yerlerinin değiştirilmesi çözümü değiştirmeyecektir. Karush-Kuhn-Tucker (KKT) tamamlayıcı koşulları aşağıdaki şekildedir:

$$\alpha_i(y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 + \xi_i) = 0 \quad \forall i \quad (3.14)$$

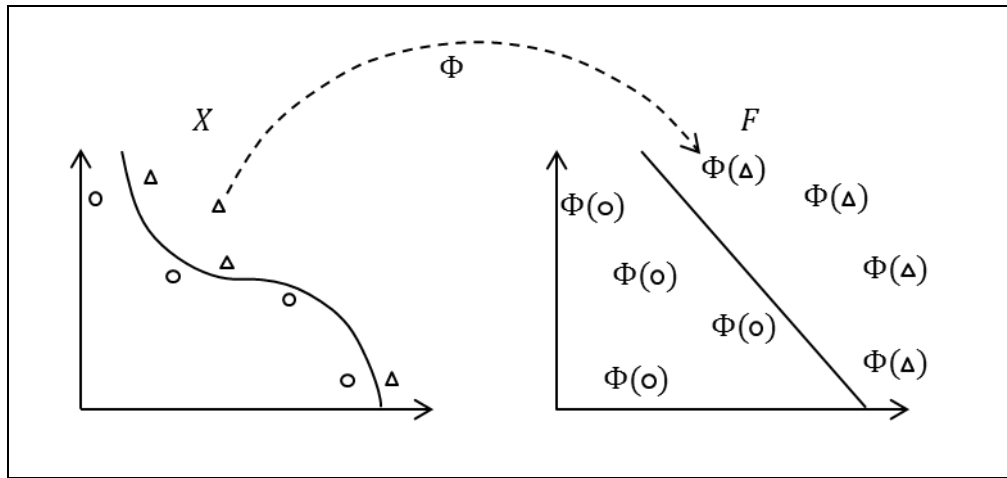
$$\beta_i \xi_i = 0 \quad \forall i \quad (3.15)$$

b 'nin hesaplanmasında KKT tamamlayıcı koşullarından faydalanılabilir. Eşitlik (3.10) ve (3.15) ile verilen eşitlikler incelendiğinde görülmektedir ki eğer $\alpha_i < C$ ise $\xi_i = 0$ olur. Dolayısıyla $0 < \alpha_i < C$ olan eğitim örnekleri Eşitlik (3.14)'te yerine konularak ($\xi_i = 0$ olacağından) b hesaplanabilir veya daha güvenilir bir sonuç için tüm mümkün b değerleri hesaplanarak ortalaması alınabilir. Yeni bir test örneği verildiğinde $sgn(\mathbf{w} \cdot \mathbf{x} + b)$ hesaplanarak bu örneğin hangi sınıfa ait olduğu bulunur.

3.1.4. Nonlinear Destek Vektör Makineleri

Doğrusal destek vektör makineleri, diskriminant fonksiyonunun verinin doğrusal fonksiyonu olmadığı durumlar için çekirdek fonksiyonunun kullanılması ile genelleştirilebilir (Boser ve diğ., 1992). Bunun için veri öncelikle girdi uzayından Şekil 3.2'de gösterildiği şekilde muhtemelen nonlinear Φ fonksiyonu yardımıyla daha yüksek boyutlu özellik uzayına (Feature space - Hilbert space) haritalanır (mapping) ve bu uzayda doğrusal sınıflandırma yapılır.

$$\Phi: \mathbb{R}^D \mapsto \mathcal{H} \quad (3.16)$$



Şekil 3.2. Özellik uzayına haritalama

Veri Φ fonksiyonu ile özellik uzayına haritalanmaktadır ki bu uzayda normalde nonlineer olan ilişkiler doğrusal olarak tanımlanabilmektedir (Cristianini ve Shawe-Taylor, 2004).

Diskriminant fonksiyonu aşağıdaki şekilde yazılabilir:

$$f(x) = \sum_{i=1}^N \alpha_i y_i \Phi(x_i) \cdot \Phi(x) + b \quad (3.17)$$

Eşitlik (3.11) ile verilen optimizasyon modelinde verinin iç çarpımlar $\langle x_i, x_j \rangle$ şeklinde yer aldığı görülmektedir. Dolayısıyla eğer aşağıdaki gibi bir çekirdek fonksiyonu var ise modeli çözmek için Φ fonksiyonunu bilmeye ve hesaplamaya gerek kalmadan, sadece çekirdek fonksiyonunu kullanmak yeterli olacaktır.

$$k(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \quad (3.18)$$

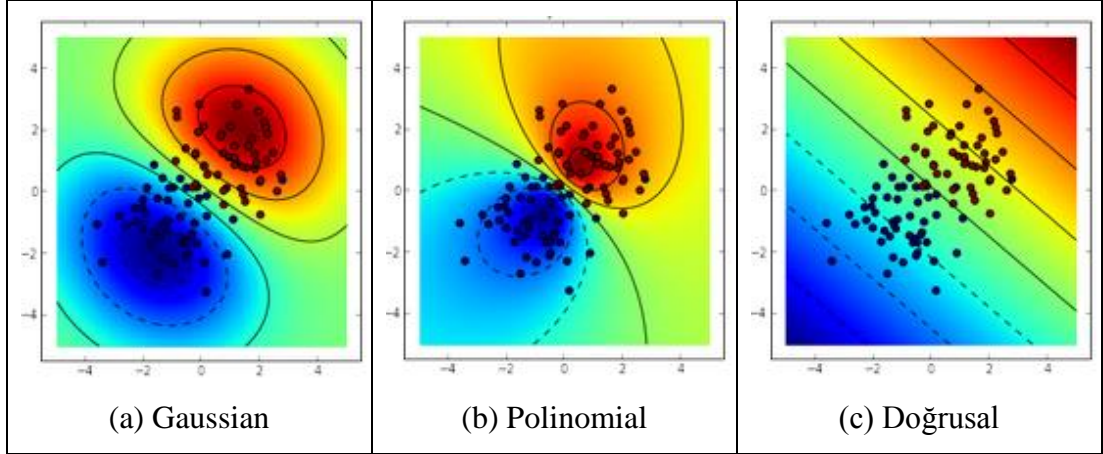
Diskriminant fonksiyonu aşağıdaki şekilde yazılabilir:

$$f(x) = \sum_{i=1}^N \alpha_i y_i k(x_i, x_j) + b \quad (3.19)$$

Nonlineer durumda optimizasyon problemi özellik uzayındaki doğrusal ayırıcı fonksiyonu bulmaktadır.

Örnek teşkil etmesi açısından bir yapay veri seti üzerinde farklı çekirdek fonksiyonları kullanılarak elde edilen ikili sınıflandırıcılar Şekil 3.3'te verilmiştir.²

² Destek vektör makinelerinde kullanılan çekirdek fonksiyonları hakkında detaylı bilgi 3'üncü Bölümde verilecektir.



Şekil 3.3. İkili sınıflandırma destek vektör makineleri

3.2. DESTEK VEKTÖR REGRESYON

Destek vektör makinelerinin regresyon problemlerinin çözümünde kullanılmasına ilişkin kapsamlı bir çalışma Smola ve Schölkopf (2004) tarafından yapılmıştır.

Eğitim örnekleri $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ şeklinde verilmiş olsun. Regresyon problemlerinde yanıt değişkeni $y_i \in \{+1, -1\}$ yerine $y_i \in \mathbb{R}$ şeklinde tanımlanmaktadır.

3.2.1. Doğrusal Destek Vektör Regresyon

ε -duyarsız (ε -insensitive) destek vektör regresyonda (ε -DVR) (Vapnik, 1995) amaç tüm eğitim verisi için gerçek yanıt değişkeni y_i 'lerden maksimum ε sapmaya sahip olan olası en düz $f(\mathbf{x})$ fonksiyonunu bulmaktır. Diğer bir deyişle hatalar ε 'dan küçük olduğu sürece önemsenmeyecek, ancak bu değerden daha büyük bir sapma kabul edilmeyecektir.

İlk olarak doğrusal durum ele alınsın. f fonksiyonu aşağıdaki şekilde ifade edilmektedir:

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b \quad \mathbf{w} \in \mathbb{R}^D, b \in \mathbb{R} \quad (3.20)$$

Eşitlik (3.20) ile verilen fonksiyonun düz olması \mathbf{w} 'nin küçük olması anlamına gelmektedir. Bunu sağlamanın bir yolu $\|\mathbf{w}\|^2$ normunu minimize etmektir.

Konveks optimizasyon problemi aşağıdaki şekilde ifade edilir:

$$\begin{aligned} \min \frac{1}{2} \|\mathbf{w}\|^2 \\ y_i - (\mathbf{w} \cdot \mathbf{x}_i + b) \leq \varepsilon \\ (\mathbf{w} \cdot \mathbf{x}_i + b) - y_i \leq \varepsilon \end{aligned} \quad (3.21)$$

3.2.2. Esnek (Yumuşak) Marjın Destek Vektör Regresyon

Bazı durumlarda bütün (\mathbf{x}_i, y_i) çiftlerini ε kesinliğinde yakınsayan f fonksiyonu mevcut olmayabilir veya bazen bazı hatalara izin verilmek istenebilir. Bunu mümkün kılmak için, esnek (yumuşak) marjın kayıp fonksiyonu kullanılarak, modele ξ^+ ve ξ^- gevşek değişkenleri dâhil edilir (Cortes ve Vapnik, 1995). Optimizasyon problemi aşağıdaki şekilde yazılır (Vapnik, 1995):

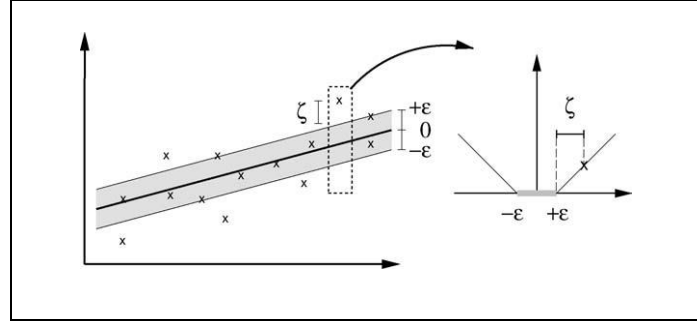
$$\begin{aligned} \min \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N (\xi_i^+ + \xi_i^-) \\ y_i - (\mathbf{w} \cdot \mathbf{x}_i + b) \leq \varepsilon + \xi_i^+ \\ (\mathbf{w} \cdot \mathbf{x}_i + b) - y_i \leq \varepsilon + \xi_i^- \\ \xi_i^+, \xi_i^- \geq 0 \end{aligned} \quad (3.22)$$

$C > 0$ sabiti f fonksiyonunun düzlüğü ve tolere edilecek hata miktarı arasında dengeyi sağlamaktadır.

ε -duyarsız kayıp fonksiyonu aşağıdaki şekilde tanımlanmaktadır:

$$|\xi|_\varepsilon = \begin{cases} 0, & |\xi| \leq \varepsilon \text{ ise} \\ |\xi| - \varepsilon, & d.d. \end{cases} \quad (3.23)$$

Şekil 3.4'te görüldüğü üzere sadece ε 'dan büyük olan hatalar ve doğrusal olarak cezalandırılmaktadır.



Şekil 3.4. Doğrusal destek vektör regresyon için esnek marjin kayıp fonksiyonu (Smola ve Schölkopf, 2004)

3.2.3. Dual Problem

Optimizasyon probleminin dual formulasyonu Lagrange çarpanları kullanılarak elde edilmektedir (Fletcher, 2000).

$$\begin{aligned}
 L = & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N (\xi_i^+ + \xi_i^-) - \sum_{i=1}^N \beta_i^+ \xi_i^+ - \sum_{i=1}^N \beta_i^- \xi_i^- \\
 & - \sum_{i=1}^N \alpha_i^+ (\varepsilon + \xi_i^+ - y_i + (\mathbf{w} \cdot \mathbf{x}_i + b)) \\
 & - \sum_{i=1}^N \alpha_i^- (\varepsilon + \xi_i^- + y_i - (\mathbf{w} \cdot \mathbf{x}_i + b))
 \end{aligned} \tag{3.24}$$

L 'nin \mathbf{w} , b , ξ_i^+ , ξ_i^- 'ye göre kısmi türevi alınırsa aşağıdaki eşitlikler elde edilir:

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) \mathbf{x}_i \tag{3.25}$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0 \tag{3.26}$$

$$\frac{\partial L}{\partial \xi_i^+} = 0 \Rightarrow C = \alpha_i^+ + \beta_i^+ \quad \forall i \tag{3.27}$$

$$\frac{\partial L}{\partial \xi_i^-} = 0 \Rightarrow C = \alpha_i^- + \beta_i^- \quad \forall i \tag{3.28}$$

Eşitlik (3.25) ile verilen ifade Eşitlik (3.24)'te yerine konulursa dual optimizasyon problemi aşağıdaki şekilde elde edilir:

$$\begin{aligned} \text{maks } & \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^+ - \alpha_i^-) (\alpha_j^+ - \alpha_j^-) \langle \mathbf{x}_i, \mathbf{x}_j \rangle - \varepsilon \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) - \sum_{i=1}^N y_i (\alpha_i^+ - \alpha_i^-) \\ & \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0 \\ & 0 \leq \alpha_i^+, \alpha_i^- \leq C \end{aligned} \quad (3.29)$$

$\mathbf{w} = \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) \mathbf{x}_i$ ifadesi yerine konulursa Eşitlik (3.20) ile verilen tahmin (prediction) fonksiyonu aşağıdaki şekilde yazılır:

$$f(\mathbf{x}) = \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) \langle \mathbf{x}_i, \mathbf{x}_j \rangle + b \quad (3.30)$$

Bu ifade destek vektör açılımı olarak adlandırılmaktadır. Yukarıdaki ifadelerden görüldüğü üzere \mathbf{w} ağırlıkları \mathbf{x}_i örneklerinin doğrusal kombinasyonu şeklinde yazılabilmektedir. Böylece destek vektörler ile temsil edilen fonksiyonun kompleksliği girdi örneklerinin boyutundan bağımsız olacak ve sadece destek vektörlerin sayısına bağlı olacaktır.

Şu noktaya da dikkat çekmek gerekir ki model \mathbf{x}_i örneklerinin çarpımları şeklinde ifade edilebilmektedir. Yani $f(\mathbf{x})$ fonksiyonunu bulurken \mathbf{w} ağırlıklarını hesaplamaya gerek kalmamaktadır. Bu özellik algoritmanın nonlinear duruma genelleştirilmesinde kullanılacaktır.

KKT tamamlayıcı koşulları aşağıdaki şekildedir:

$$\begin{aligned} \alpha_i^+ (\varepsilon + \xi_i^+ - y_i + (\mathbf{w} \cdot \mathbf{x}_i + b)) &= 0 \\ \alpha_i^- (\varepsilon + \xi_i^- - (\mathbf{w} \cdot \mathbf{x}_i + b) + y_i) &= 0 \end{aligned} \quad (3.31)$$

$$\begin{aligned} (C - \alpha_i^+) \xi_i^+ &= 0 \\ (C - \alpha_i^-) \xi_i^- &= 0 \end{aligned} \quad (3.32)$$

Yukarıdaki ifadelerden yola çıkılarak faydalı çıkarımlar yapılabilir. Öncelikle sadece $\alpha_i^+, \alpha_i^- = C$ olan (\mathbf{x}_i, y_i) örnekleri ε -duyarsız tüpün dışında kalacaktır. İkinci olarak

$\alpha_i^+ \alpha_i^- = 0$, yani hiç bir α_i^+, α_i^- dual değişken çifti aynı zamanda sıfırdan farklı olamaz.

Eşitlik (3.31) ile verilen ifadeye dayanılarak söylenebilir ki sadece $|f(\mathbf{x}_i) - y_i| \geq \varepsilon$ olduğu durumda Lagrange çarpanları sıfırdan farklı olacaktır. Diğer bir deyişle Şekil 3.4'te gösterilen ε -tüpün içindeki örnekler için α_i^+, α_i^- sifira eşit olacaktır ($|f(\mathbf{x}_i) - y_i| < \varepsilon$ olduğunda Eşitlik (3.31) ile verilen ifadede ikinci çarpan sıfırdan farklı olacaktır ve KKT koşullarının sağlanması için α_i^+, α_i^- sıfır olmalıdır.). Dolayısıyla \mathbf{w} 'yi hesaplariken tüm \mathbf{x}_i 'leri kullanmamıza gerek kalmamaktadır. Katsayıları sıfırdan farklı olan bu örnekler destek vektörler olarak adlandırılır.

3.2.4. Nonlinear Destek Vektör Regresyon

Destek vektör regresyonun nonlinear duruma genelleştirilmesi için \mathbf{x}_i örnekleri $\Phi: \mathbb{R}^D \mapsto \mathcal{H}$ fonksiyonu yardımı ile özellik uzayına haritalanmaktadır. Destek vektör algoritmaları sadece \mathbf{x}_i örneklerinin iç çarpımlarına bağlı olduğundan Φ fonksiyonunu bilmemize gerek kalmadan $k(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$ fonksiyonunu hesaplamak yeterli olacaktır. Optimizasyon problemi aşağıdaki şekilde yazılır (Smola ve Schölkopf, 2004):

$$\begin{aligned} \text{maks } & \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-)k(\mathbf{x}_i, \mathbf{x}_j) - \varepsilon \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) - \sum_{i=1}^N y_i (\alpha_i^+ - \alpha_i^-) \\ & \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0 \\ & 0 \leq \alpha_i^+, \alpha_i^- \leq C \end{aligned} \quad (3.33)$$

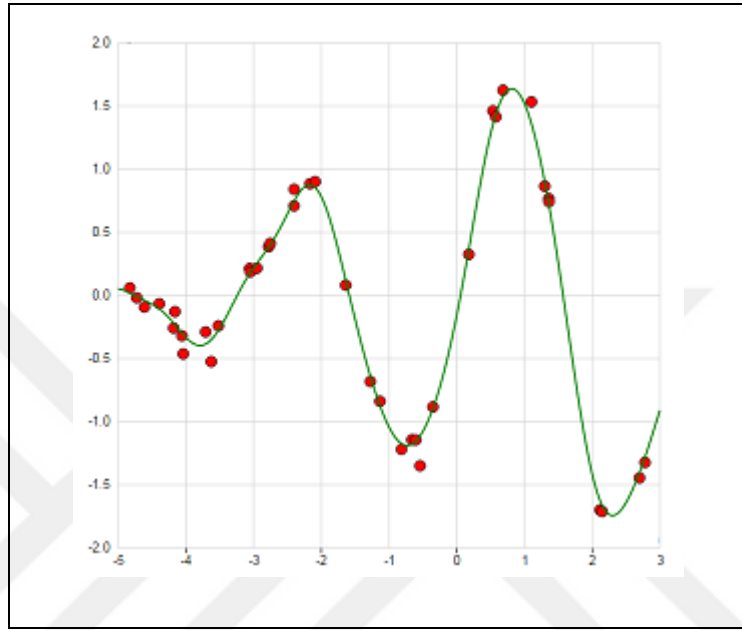
Ağırlıklar ve tahmin fonksiyonu aşağıdaki şekilde yazılır:

$$\mathbf{w} = \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) \Phi(\mathbf{x}_i) \quad (3.34)$$

$$f(\mathbf{x}) = \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-)k(\mathbf{x}_i, \mathbf{x}_j) + b \quad (3.35)$$

Doğrusal durumdan farkı w ağırlıklarının açık olarak elde edilmemesidir. Bununla birlikte nonlinear durumda optimizasyon problemi girdi uzayında değil özellik uzayındaki en düz fonksiyonu bulmaktır.

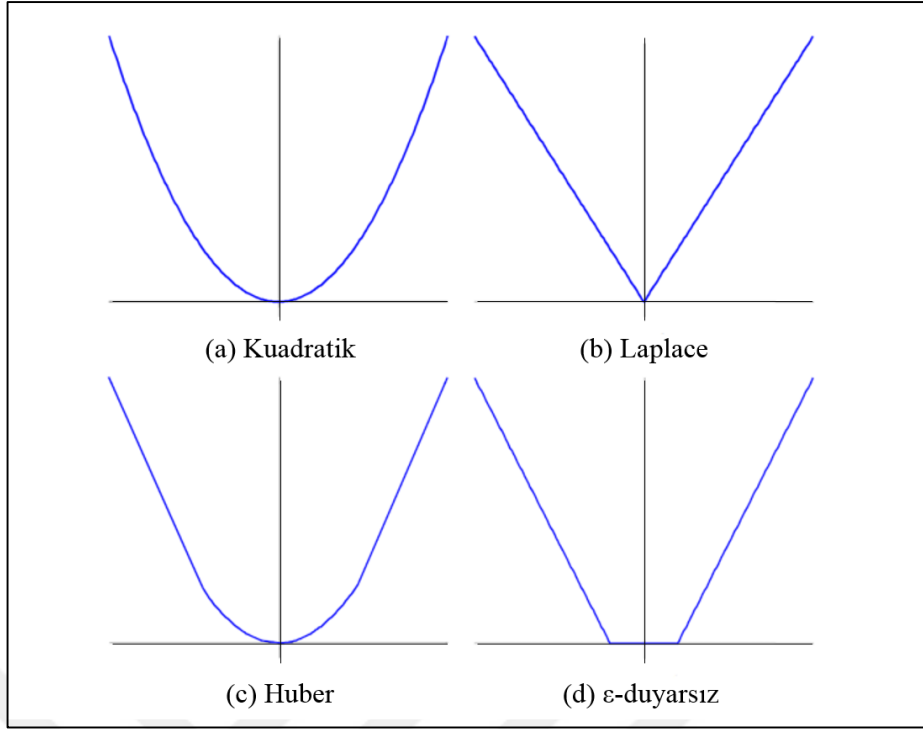
Bir yapay veri seti üzerinde Gaussian çekirdek fonksiyonu kullanılarak elde edilen destek vektör regresyon modeli örnek teşkil etmesi açısından Şekil 3.5'te verilmiştir.



Şekil 3.5. Gaussian çekirdek fonksiyonu ile destek vektör regresyon

3.2.5. Kayıp Fonksiyonları

Destek vektör makinelerinin regresyon problemlerine uygulanabilmesi için modele kayıp fonksiyonu dâhil edilmektedir. Standart olarak kullanılan kayıp fonksiyonu önceki kısımlarda açıklandığı üzere ϵ -duyarsız kayıp fonksiyonudur. Şekil 3.6'da yaygın olarak kullanılan kayıp fonksiyonları verilmiştir.

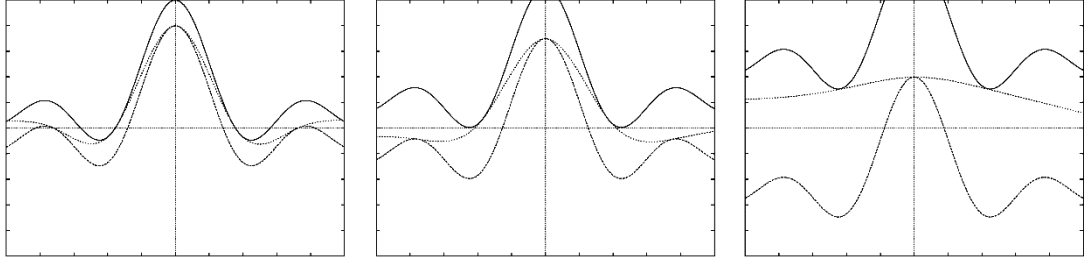


Şekil 3.6. Kayıp Fonksiyonları (Gunn, 1998)

Şekil 3.6 (a)'da verilen kayıp fonksiyonu klasik en küçük kareler hata kriterine karşılık gelmektedir. Şekil 3.6 (b)'de verilen Laplace kayıp fonksiyonu aykırı değerlere karşı kuadratik kayıp fonksiyonuna göre daha az duyarlıdır. Huber (1964) tarafından Şekil 3.6 (c)'de verilen sağlam (robust) ve verinin dağılımının bilinmediği durumlarda optimal özelliklere sahip olan kayıp fonksiyonu önerilmiştir. Bu üç kayıp fonksiyonunun destek vektör sayısının küçük tutulmasında bir katkısı bulunmamaktadır. Bu konuya çözüm olarak Vapnik (1995) tarafından destek vektör sayısının az olmasını (sparseness) sağlamak amacıyla Huber kayıp fonksiyonuna benzer şekilde Şekil 3.6 (d)'de verilen ϵ -duyarsız kayıp fonksiyonu önerilmiştir.

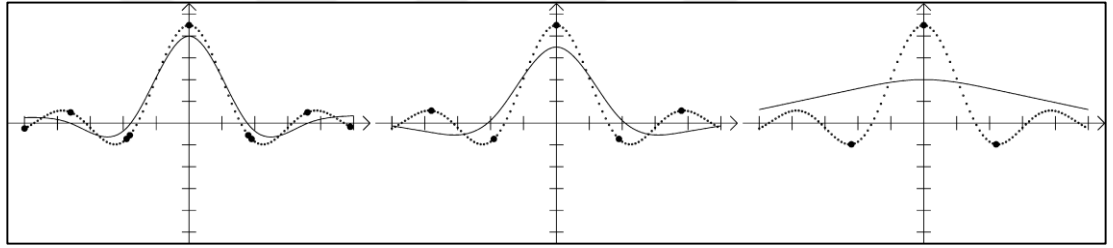
3.2.6. Duyarlılık Parametresinin Seçimi

Destek vektör regresyon verilen kesinlik derecesinde veriyi tahmin eden fonksiyonlar arasında en düz olanını seçmektedir. Şekil 3.7'de $\text{sinc}(x)$ fonksiyonunun sırasıyla soldan sağa $\epsilon = 0.1, 0.2$ ve 0.3 kesinlik dereceleri ile kestirimleri verilmektedir. Üst ve alt çizgiler ϵ tüpünün genişliğini göstermektedir. Ortadaki kesikli çizgi ise regresyon kestirimidir.



Şekil 3.7. Duyarlılık parametresinin regresyon kestirimine etkisi (Smola ve Schölkopf (2004))

Regresyon kestirimi için belirlenen kesinlik derecesi azaldıkça gerekli olan destek vektör (DV) sayısı da azalacaktır. Şekil 3.8’de duyarlılık parametresinin destek vektör sayısı üzerindeki etkisi gösterilmektedir. Yine $\text{sinc}(x)$ fonksiyonunun sırasıyla soldan sağa $\varepsilon = 0.1, 0.2$ ve 0.3 kesinlik dereceleri ile kestirimleri verilmektedir. Kesintisiz çizgi regresyonu, küçük noktalar veri noktalarını, büyük noktalar destek vektörleri göstermektedir.



Şekil 3.8. Duyarlılık parametresinin destek vektör sayısı üzerindeki etkisi (Smola ve Schölkopf (2004))

3.2.7. Otomatik Hata Kontrolü

Destek vektör regresyon yönteminde iyi performans elde edilmesi için ε parametresinin uygun şekilde seçimi önemli bir problem sahası teşkil etmektedir.

Bu parametrenin optimal değerinin bulunması için Schölkopf ve diğ. (2000) tarafından ε parametresinin ve ε parametresini minimize edecek ekstra bir terimin amaç fonksiyonuna dâhil edildiği otomatik doğruluk kontrol yöntemi önerilmiştir ve bu yöntem ν -destek vektör regresyon (ν -DVR) olarak adlandırılmıştır.

ν -destek vektör regresyonda hata marjini esnek bir tüp oluşturmakta ve veriyle birlikte dinamik olarak değişmektedir. ν -destek vektör regresyon için primal problem aşağıdaki şekilde yazılmaktadır:

$$\min_{\mathbf{w}, b, \xi^+, \xi^-, \varepsilon} \frac{1}{2} \|\mathbf{w}\|^2 + C \left(v\varepsilon + \frac{1}{N} \sum_{i=1}^N (\xi_i^+ + \xi_i^-) \right)$$

$$y_i - (\mathbf{w}^\top \Phi(\mathbf{x}_i) + b) \leq \varepsilon + \xi_i^+$$

$$(\mathbf{w}^\top \Phi(\mathbf{x}_i) + b) - y_i \leq \varepsilon + \xi_i^-$$

$$\varepsilon > 0$$

$$\xi_i^+, \xi_i^- \geq 0, \quad i = 1, \dots, N \quad (3.36)$$

Burada ε hata genişliği, $v \in (0,1)$ aracılığı ile model kompleksliği ve gevşek değişkenler arasında dengeyi sağlamaktadır. Dual problem aşağıdaki şekilde yazılabilir:

$$\max_{\alpha^+, \alpha^-} -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^+ - \alpha_i^-) (\alpha_j^+ - \alpha_j^-) k(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^N y_i (\alpha_i^+ - \alpha_i^-)$$

$$\sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0$$

$$\sum_{i=1}^N (\alpha_i^+ + \alpha_i^-) \leq Cv$$

$$\alpha_i^+, \alpha_i^- \in [0, C/N], \quad i = 1, \dots, N \quad (3.37)$$

Kullanıcılar C sabitini genellikle 1 gibi küçük bir sayıya eşit olarak seçtiklerinden dolayı, C/N çok küçük olabilmektedir. Bu yüzden LibSVM (A Library for Support Vector Machines) (Chang ve Lin, 2011) yazılımında kullanıcı tarafından belirlenen düzleştirme parametresi (regularization parameter) C/N olarak alınmaktadır. Yani kullanıcı tarafından belirlenen parametre $\bar{C} = C/N$ 'dir.

LibSVM aşağıdaki problemi çözmektedir:

$$\max_{\alpha^+, \alpha^-} -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-)k(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^N y_i (\alpha^+ - \alpha^-)$$

$$\sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0$$

$$\sum_{i=1}^N (\alpha_i^+ + \alpha_i^-) \leq CNv$$

$$\alpha_i^+, \alpha_i^- \in [0, C], \quad i = 1, \dots, N \quad (3.38)$$

Chang ve Lin (2002), ε -DVR modelinin (\bar{C}, ε) parametreleriyle ve v -DVR modelinin $(N\bar{C}, v)$ parametreleriyle çözülmesi durumunda aynı sonucun elde edileceğini göstermiştir.

ε parametresinin otomatik olarak belirlenmesinin yanı sıra v -destek vektör regresyon modelinin bir avantajı daha bulunmaktadır. v parametresi ile destek vektörlerin sayısı kontrol edilmektedir. v parametre değeri hataların oranı için bir üst sınır ve destek vektörlerin oranı için bir alt sınırdır (Schölkopf ve diğ., 2000).

3.3. TEK SINIF DESTEK VEKTÖR MAKİNELERİ

Destek vektör makineleri anomali tespiti problemlerinin çözümünde de kullanılabilir. Dağılım tahmini (distribution estimation) için tek sınıf destek vektör makineleri Schölkopf ve diğ. (2001) tarafından önerilmiştir. Eğitim örnekleri $\mathbf{x}_i \in \mathbb{R}^D, i = 1, \dots, N$ şeklinde verilmektedir. Örneklere ait sınıf bilgisi mevcut değildir.

Primal optimizasyon problemi aşağıdaki şekildedir:

$$\min \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i + b$$

$$\mathbf{w}^\top \Phi(\mathbf{x}_i) + b + \xi_i \geq 0 \quad \forall i$$

$$\mathbf{w} \in \mathbb{R}^D, \quad \xi_i \geq 0, \quad b \in \mathbb{R} \quad (3.39)$$

Primal problemin Lagrange duali aşağıdaki şekilde elde edilmektedir:

$$L = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i + b - \sum_{i=1}^N \alpha_i (w^T \Phi(x_i) + b + \xi_i) - \sum_{i=1}^N \beta_i \xi_i \quad (3.40)$$

L 'nin primal parametrelere göre türevi alınır aşağıdaki eşitlikler elde edilir:

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^N \alpha_i \Phi(x_i) \quad (3.41)$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^N \alpha_i = 0 \quad (3.42)$$

$$\frac{\partial L}{\partial \xi_i} = 0 \Rightarrow C = \alpha_i + \beta_i \quad \forall i \quad (3.43)$$

Dual formülasyon aşağıdaki şekilde elde edilmektedir:

$$\begin{aligned} & maks - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j k(x_i, x_j) \\ & \sum_{i=1}^N \alpha_i = 1 \\ & 0 \leq \alpha_i \leq C \quad \forall i \end{aligned} \quad (3.44)$$

Diskriminant fonksiyonu aşağıdaki şekilde yazılır:

$$f(x) = \sum_{i=1}^N \alpha_i k(x_i, x_j) + b \quad (3.45)$$

3.4. ÇOKLU GRUP SINIFLANDIRMA DESTEK VEKTÖR MAKİNELERİ

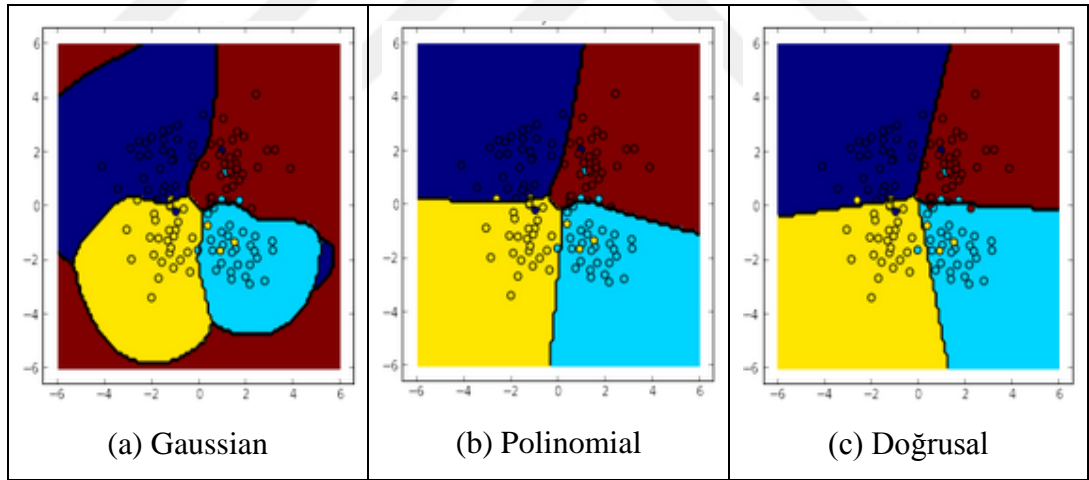
Çoklu grup sınıflandırma problemlerinde eğitim setindeki x_i örneği $k > 2$ gruptan birisine aittir ve grup etiketleri $y_i \in \{1, \dots, k\}$ şeklinde verilmektedir.

Destek vektör makineleri yönteminin çoklu grup sınıflandırma problemleri için geliştirilmesi konusunda önerilen farklı yaklaşımlar mevcuttur. Yaygın olarak kabul edilen yaklaşım çoklu grup probleminin çok sayıda ikili sınıflandırma problemine

indirgenmesidir. Bu yaklaşımda kullanılan iki yöntem söz konusudur. Bire-karşı-hepsi (one-versus-all) yönteminde bir grubu diğer gruplardan ayırmak için kullanılan k farklı sınıflandırıcı oluşturulmaktadır. Test aşamasında maksimum çıktı değerine sahip olan ikili sınıflandırıcı test örneğinin grubunu belirlemektedir. Bire-bir (one-versus-one) yönteminde ise olası bütün grup çiftleri için $k(k - 1)/2$ adet sınıflandırıcı oluşturulmaktadır. Test aşamasında sınıflandırma için oylama stratejisi uygulanmaktadır. Bu stratejide her sınıflandırıcı test örneğini iki gruptan birisine atamaktadır ve atanan gruba ait oy bir arttırılmaktadır. Sonuçta en çok oy alan grup örneğin grubunu belirlemektedir.

Çoklu grup sınıflandırma probleminin çözümü için, çok sayıda ikili sınıflandırma problemlerine ayrıştırılması yerine tek bir optimizasyon probleminin çözülmesine dayanan yöntemler de mevcuttur. Önerilen farklı yöntemlerin karşılaştırılması konusunda kapsamlı bir çalışma Hsu ve Lin (2002) tarafından yapılmıştır.

Bir yapay veri seti üzerinde bire-karşı-hepsi yöntemi ile farklı çekirdek fonksiyonları kullanılarak elde edilen sonuçlar örnek teşkil etmesi açısından Şekil 3.9’da verilmiştir.



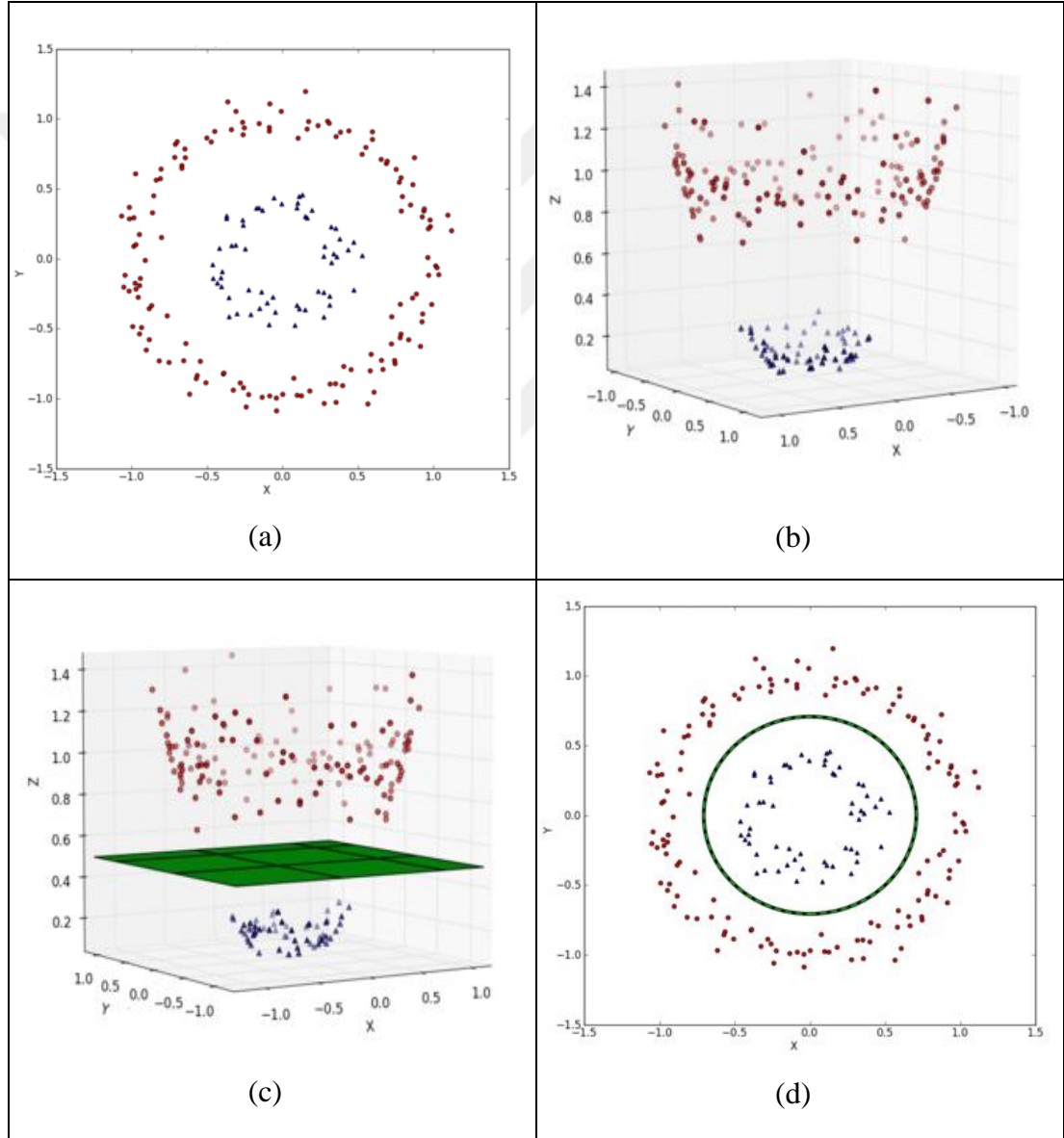
Şekil 3.9. Çoklu grup sınıflandırma destek vektör makineleri

3.5. ÇEKİRDEK YÖNTEMİ

Çekirdek fonksiyonları benzerlik ölçüsü olarak kullanılmaktadır. Çekirdek yöntemi ile ilgili detaylı çalışma Cristianini ve Shawe-Taylor (2000 ve 2004) tarafından yapılmıştır.

3.5.1. Yüksek Boyutlu Uzaya Haritalama

Bu kısımda çekirdek yönteminin basit bir örnek üzerinde görselleştirilmesi amaçlanmıştır. Şekil 3.10 (a)'de \mathbb{R}^2 uzayında doğrusal olarak ayrılamayan veri seti görülmektedir. Bu veri seti $(x_1, x_2) \mapsto (x_1, x_2, x_1^2 + x_2^2)$ şeklinde \mathbb{R}^3 uzayına haritalandığında Şekil 3.10 (b) elde edilmektedir. Şekil 3.10 (c)'de görüldüğü üzere veri seti \mathbb{R}^3 uzayında bir hiperdüzlem ile doğrusal olarak ayrılabilir. Şu konuya dikkat çekmek gerekir ki Şekil 3.10 (d) ile verilen \mathbb{R}^2 uzayına geri yansıtılan ayrırıcı düzlem nonlineerdir.



Şekil 3.10. Yüksek boyutlu uzaya haritalama (Kim, 2013)

3.5.2. Çekirdek Matrisi

Çekirdek yöntemlerinin çözümü iki aşamadan oluşmaktadır. Birinci aşama özellik uzayına haritalama, ikinci aşama ise bu uzayda doğrusal ilişkileri tespit etmek için öğrenme algoritmalarının uygulanmasıdır.

Çekirdek yönteminin en önemli özelliği algoritmaların sadece örneklerin iç çarpımlarını içerecek şekilde formüle edilmesidir. Böylece iç çarpımlar orijinal veriler kullanılarak çekirdek fonksiyonu ile etkin bir şekilde hesaplanabilir. Verilen bir eğitim seti ve çekirdek fonksiyonu için çekirdek matrisi, diğer bir deyişle Gram matrisi, hesaplanır.

K	1	2	...	N
1	$k(\mathbf{x}_1, \mathbf{x}_1)$	$k(\mathbf{x}_1, \mathbf{x}_2)$...	$k(\mathbf{x}_1, \mathbf{x}_N)$
2	$k(\mathbf{x}_2, \mathbf{x}_1)$	$k(\mathbf{x}_2, \mathbf{x}_2)$...	$k(\mathbf{x}_2, \mathbf{x}_N)$
⋮	⋮	⋮	⋮	⋮
N	$k(\mathbf{x}_N, \mathbf{x}_1)$	$k(\mathbf{x}_N, \mathbf{x}_2)$...	$k(\mathbf{x}_N, \mathbf{x}_N)$

Şekil 3.11. Çekirdek matrisi

Bu matris tüm örnek çiftlerinin çekirdek fonksiyonu değerlerinden oluşmaktadır ve bilgi geçidi (information bottleneck) olarak görev yapmaktadır. Çünkü çekirdek algoritmaları ve öğrenme aşamasının uygulanması için gerekli olan dağılım, model, gürültü vb. hakkında tüm bilgiler bu matristen elde edilmektedir. Öğrenme algoritmasının özellik uzayının seçimi, model seçimi ve aslında eğitim verisinin kendisi hakkında bilgi edinebileceği tek unsur çekirdek matrisidir. Bu nedenle birçok alternatif arasından en iyi çekirdek fonksiyonunu seçmek çekirdek tabanlı algoritmaların uygulanmasında en kritik aşamadır ve çekirdek matrisi öğrenen sistemin genelleştirme performansının değerlendirilmesinde kullanılabilir.

3.5.3. Mercer Koşulları

Önceki bölümlerde açıklanan özellikte haritalama fonksiyonu ve özellik uzayına sahip olan çekirdek fonksiyonlarının özellikleri, yani hangi çekirdek fonksiyonlarının özellik uzayındaki iç çarpımlara karşılık geldiği, Mercer koşulları ile tanımlanmaktadır.

$k(\mathbf{x}, \mathbf{y}) = \sum_i \Phi(\mathbf{x})_i \Phi(\mathbf{y})_i$ olmak üzere, her $g(\mathbf{x})$ için, $\int g(\mathbf{x})^2 dx$ sonlu ve, $\int k(\mathbf{x}, \mathbf{y}) g(\mathbf{x}) g(\mathbf{y}) dx dy \geq 0$ olmalıdır (Mercer, 1909).

3.5.4. Çekirdek Fonksiyonları

Destek vektör makineleri için yaygın olarak kullanılan çekirdek fonksiyonları aşağıda verilmiştir (Cristianini ve Shawe-Taylor, 2004):

$$\text{Doğrusal çekirdek fonksiyonu} \quad : k_D(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j \quad (3.46)$$

$$\text{Polinomial çekirdek fonksiyonu} \quad : k_P(\mathbf{x}_i, \mathbf{x}_j) = ((\mathbf{x}_i \cdot \mathbf{x}_j) + 1)^p \quad p \in \mathbb{N} \quad (3.47)$$

$$\begin{aligned} \text{Gaussian çekirdek fonksiyonu} \quad : k_G(\mathbf{x}_i, \mathbf{x}_j) &= e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2} \quad \sigma > 0 \\ &= e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2} \quad \gamma = \frac{1}{2\sigma^2} \end{aligned} \quad (3.48)$$

3.6. OPTİMİZASYON ALGORİTMALARI

Bu bölümde destek vektör makinelerinin uygulamasında kullanılan optimizasyon algoritmalarına yer verilecektir.

3.6.1. İç Noktalar Algoritmaları

İç noktalar (interior points) algoritmaları yönteminde optimizasyon probleminin duali hesaplanmakta, primal ve dual problem eş zamanlı olarak çözülmektedir. İteratif olarak KKT koşulları kullanılarak mümkün bir çözüm bulunmaktadır. Elde edilen değişken setinin uygunluğunun değerlendirilmesinde primal ve dual amaç fonksiyonları arasındaki dualite boşluğundan faydalanılmaktadır. Mümkün primal ve dual değişkenler için konveks minimizasyon probleminin primal amaç fonksiyonu her zaman için dual amaç fonksiyonuna eşit veya büyük olmaktadır. Destek vektör makineleri sadece doğrusal kısıtlara sahip olduğundan güçlü dualite teoreminin şartları sağlanmaktadır ve optimal çözümde fark sifıra eşit olmaktadır. Bu yüzden dualite boşluğu (farkı), mevcut değişken setinin çözüme ne kadar yakın olduğunun ölçüsü olarak kullanılmaktadır (Vanderbei, 1998).

3.6.2. Alt Küme Seçme Algoritmaları

Büyük veri setlerinde hafıza ve işlemci limitlerinden dolayı çekirdek matrisini hesaplamak ve hafızada tutmak zor olmaktadır. Bu problem için Vapnik (1982) tarafından önerilen yöntem çözümün sadece destek vektörlere bağlı olmasına dayanmaktadır. Eğer destek vektör kümesi önceden biliniyorsa ve bu kümenin boyutu hafızaya uygunsuzsa direkt olarak indirgenmiş problem çözülebilir. Bu yaklaşımdaki sorun problem çözülmeden önce destek vektörlerin bilinmemesidir. Çözüm hafıza kısıtlarına uyan keyfi bir alt küme, ilk yığın (chunk) ile başlamaktır. Destek vektör algoritmaları bu alt küme üzerinde eğitilir, elde edilen destek vektörler saklanır ve mevcut tahmin edicinin hata yaptığı (regresyon problemleri için hata tüpünün dışında kalan) veriler yığına dâhil edilir. Sistem yeniden eğitilir ve tüm KKT koşulları sağlanıncaya kadar bu prosedüre devam edilir.

Osuna ve diğ. (1997b) değişkenlerin sadece bir alt kümesini çalışma seti olarak kullanmayı önermiştir. Bu yöntemde diğer değişkenler sabit tutulmakta, dondurulmakta (freezing), problem çalışma setine göre optimize edilmektedir.

3.6.3. Sıralı Minimal Optimizasyon

Platt (1999) yığın (chunking) yöntemini genişletmiş ve Sıralı Minimal Optimizasyon (Sequential Minimal Optimization - SMO) yöntemini önermiştir. Bu yöntemde iteratif olarak sadece iki elemanlı alt kümeler seçilmekte ve hedef fonksiyonu bu alt kümelere göre optimize edilmektedir. Bu yöntem iyi yakınsama özelliği göstermekte ve uygulama kolaylığı sağlamaktadır. Anahtar nokta iki elemanlı çalışma seti için optimizasyon alt probleminin kuadratik optimizasyon çözücülerine ihtiyaç duyulmaksızın analitik olarak çözülebilmesidir.

Destek vektör algoritmalarında aşağıdaki kısıtlar mevcut idi:

$$0 \leq \alpha_i \leq C \quad (3.49)$$

$$\sum_{i=1}^N \alpha_i y_i = 0 \quad (3.50)$$

Bu kısıtlar iki çarpan için aşağıdaki şekilde indirgenir:

$$0 \leq \alpha_1, \alpha_2 \leq C \quad (3.51)$$

$$y_1 \alpha_1 + y_2 \alpha_2 = k \quad (3.52)$$

İndirgenmiş problem analitik olarak çözülebilir. Burada k eşitlik kısıtında yer alan diğer terimlerin toplamlarının negatifidir ki bu terimler her iterasyonda sabit tutulmaktadır.

Algoritma şu şekilde devam etmektedir:

i. KKT koşullarını sağlamayan bir α_1 Lagrange çarpanı seçilir.

ii. İkinci çarpan α_2 seçilir ve (α_1, α_2) çifti optimize edilir.

iii. Yakınsama sağlanıncaya kadar *i* ve *ii* adımları tekrar edilir.

Bütün Lagrange çarpanları KKT koşullarını sağladığında problem çözülmüş olur (Platt, 1999).

4. ÇOKLU ÇEKİRDEK ÖĞRENME ALGORİTMALARI

Destek vektör makinelerinde çekirdek fonksiyonunun kullanılması ile karar fonksiyonu, $\{\mathbf{x}_i\}_{i=1}^N$ eğitim örnekleri ve \mathbf{x} test örneğinin çekirdek fonksiyonu değerleri cinsinden aşağıdaki şekilde ifade edilmektedir (Burges, 1998).

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b \quad (4.1)$$

Çekirdek fonksiyonu $k(\cdot, \cdot)$, yaygın olarak kullanılan doğrusal, polinomial veya Gaussian olabileceği gibi çözülmek istenen probleme özgü önerilmiş farklı çekirdek fonksiyonları da mevcuttur. Destek vektör makineleri yönteminde en önemli problem sahalarından birisi model seçimi, yani çekirdek fonksiyonunun ve ilgili parametrelerin (örneğin polinomial çekirdek fonksiyonunun derecesi, Gaussian çekirdek fonksiyonunda varyans gibi) seçimidir. Genel olarak parametre seçimi için çapraz geçerlilik yöntemi kullanılmakta, çekirdek fonksiyonu ise deneme-yanılma yöntemi ile seçilmektedir.

Bu problemin çözümü için tek bir çekirdek fonksiyonunun ve ilgili parametrelerinin kullanılması yerine çoklu çekirdek fonksiyonlarının kullanılmasını öneren çoklu çekirdek öğrenme algoritmaları geliştirilmiştir. Çoklu çekirdek öğrenme algoritmaları için kapsamlı bir çalışma Gönen ve Alpaydın (2011) tarafından yapılmıştır. Çoklu çekirdek fonksiyonu aşağıdaki şekilde ifade edilmektedir:

$$k_\eta(\mathbf{x}_i, \mathbf{x}_j) = f_\eta \left(\{k_m(\mathbf{x}_i^m, \mathbf{x}_j^m)\}_{m=1}^P \right) \quad (4.2)$$

Burada $f_\eta: \mathbb{R}^P \rightarrow \mathbb{R}$ kombinasyon fonksiyonudur, doğrusal veya nonlinear fonksiyon olabilir. Çekirdek fonksiyonları, $\{k_m: \mathbb{R}^{D_m} \times \mathbb{R}^{D_m} \rightarrow \mathbb{R}\}_{m=1}^P$ veri örneklerinin P özellik gösterimini alırlar. $\mathbf{x}_i^m \in \mathbb{R}^{D_m}$ ve D_m ilgili özellik uzayının boyutu olmak üzere veri örnekleri $\mathbf{x}_i = \{\mathbf{x}_i^m\}_{m=1}^P$ şeklinde ifade edilir. η kombinasyon fonksiyonunu parametrelili hale getirmektedir (Gönen ve Alpaydın, 2011).

Farklı çekirdek fonksiyonları farklı benzerlik ölçülerini temsil etmektedir. Hangi çekirdek fonksiyonunun çözülmek istenen problem için en iyi sonucu vereceğinin deneyerek bulunması yerine çoklu çekirdek öğrenme algoritmaları en iyi çekirdek fonksiyonunu seçmekte veya çekirdek fonksiyonlarının bir kombinasyonunu kullanmaktadır. Bununla birlikte çoklu çekirdek fonksiyonları farklı kaynaklardan gelen ve farklı gösterimlere sahip olan girdi değişkenlerinin kullanılmasına da olanak sağlamaktadır.

4.1. ÇOKLU ÇEKİRDEK ALGORİTMALARININ SINIFLANDIRILMASI

Konunun daha iyi anlaşılmasının sağlanması amacıyla öncelikle çoklu öğrenme algoritmaları için genel çerçeve çizilecektir. Bu aşamada Gönen ve Alpaydın (2011) tarafından yapılan sınıflandırma çalışmasından faydalanılmıştır. Önerilen çoklu öğrenme algoritmaları çekirdek diskriminant analizi, çekirdek ridge regresyon, Gaussian süreç gibi farklı çekirdek tabanlı öğrenme yöntemlerini de kapsamaktadır. Takip eden kısımda destek vektör regresyon için önerilen çoklu çekirdek öğrenme algoritmaları hakkında detaylı bilgi sunulacaktır.

Kombinasyon fonksiyonunu belirlemek için kullanılan öğrenme yöntemine göre:

Sabit kurallar: Parametresiz fonksiyonlardır ve dolayısıyla eğitim aşamasına gerek duymazlar (çekirdek fonksiyonlarının toplamları veya çarpımları gibi).

Sezgisel yaklaşımlar: Parametrelili kombinasyon fonksiyonlarıdır ve fonksiyonun parametreleri genellikle çekirdek fonksiyonlarının tek başına kullanıldıklarında gösterdikleri performans değerlerine göre belirlenir.

Optimizasyon yaklaşımları: Parametrelili kombinasyon fonksiyonlarıdır ve fonksiyonun parametreleri optimizasyon probleminin çözülmesi ile öğrenilir.

Bayesyen yaklaşımlar: Kombinasyon parametrelerini belirlemede önsel bilgidan faydalanılır.

Boosting yaklaşımlar: Topluluk (ensemble) yöntemlerinden esinlenerek performans iyileşmesi duruncaya kadar her iterasyonda yeni bir çekirdek fonksiyonu modele dâhil edilir.

Çekirdek kombinasyon fonksiyonuna göre:

Doğrusal kombinasyon yöntemleri: Ağırlıksız toplam (çekirdeklerin toplamlarının veya ortalamalarının birleştirilmiş çekirdek fonksiyonu olarak kullanılması) veya ağırlıklı toplam yöntemleri mevcuttur.

Ağırlıklı toplam için önerilen yöntemler η ağırlıkları için uyguladıkları kısıtlar bakımından farklılık göstermektedir.

$$\text{Doğrusal kombinasyon} \quad : \eta \in \mathbb{R}^P \quad (4.3)$$

$$\text{Konik kombinasyon} \quad : \eta \in \mathbb{R}_+^P \quad (4.4)$$

$$\text{Konveks kombinasyon} \quad : \eta_m \in \mathbb{R}_+^P \quad \text{ve} \quad \sum_{m=1}^P \eta_m = 1 \quad (4.5)$$

Konik ve konveks toplamların yorumlama açısından bir avantajı bulunmaktadır. Pozitif çekirdek ağırlıkları birleştirilen çekirdek fonksiyonlarının göreceli önemleri hakkında bilgi vermektedir.

Kombinasyon parametreleri üzerinde ilave kısıtlar da uygulanabilir (çekirdek ağırlıkları için ℓ_p norm veya çekirdek matrisi için iz kısıtı gibi).

Nonlinear kombinasyon yöntemleri: Çekirdeklerin nonlinear fonksiyonlarını kullanırlar (çarpım, üs gibi).

Veriye özgü kombinasyon yöntemleri: Her bir veri örneği için farklı çekirdek ağırlıkları atarlar. Böylece verideki yerel dağılımlar belirlenebilir ve farklı girdi bölgelerine uygun çekirdek fonksiyonları öğrenilebilir.

Kombinasyon fonksiyonunun parametreleri seçilirken optimize edilen amaç fonksiyonuna göre:

Benzerlik-tabanlı fonksiyonlar: Birleştirilmiş çekirdek matrisi ile eğitim setinden hesaplanan optimum çekirdek matrisi arasında bir benzerlik ölçüsü hesaplanır. Kombinasyon fonksiyonunun parametreleri benzerlik maksimize edilecek şekilde seçilir. İki çekirdek matrisi arasındaki benzerlik çekirdek uyuşması (kernel alignment), Öklid uzaklığı (Euclidean distance), Kullback-Leibler uzaklığı (divergence) gibi benzerlik ölçüleri kullanılarak hesaplanabilir.

Yapısal risk fonksiyonları: Yapısal risk minimizasyonu prensibini temel alırlar. Modelin performansını gösteren model kompleksliği veya hata terimi gibi düzleştirme terimlerinin (regularization term) toplamı minimize edilir.

Bayesyen fonksiyonlar: Bayesyen formülasyon kullanılarak aday çekirdeklerden oluşturulan birleştirilmiş çekirdek fonksiyonunun uygunluğu ölçülür. En çok olabilirlik tahmini veya maksimum sonsal (posteriori) tahmin maksimize edilir.

Eğitim yöntemine göre:

Tek aşamalı yöntemler: Kombinasyon parametreleri ve birleştirilen fonksiyonların parametreleri tek aşamada hesaplanır. Ardışık veya eş zamanlı yaklaşımlar kullanılabilir. Ardışık yaklaşımda önce kombinasyon fonksiyonu parametreleri belirlenir, daha sonra model birleştirilmiş çekirdek fonksiyonu ile eğitilir. Eş zamanlı yaklaşımda her iki parametre seti birlikte öğrenilir.

İki aşamalı yöntemler: İteratif bir yaklaşım kullanır. Her iterasyonda önce model parametreleri sabit tutularak kombinasyon fonksiyonu parametreleri güncellenir, daha sonra kombinasyon fonksiyonu parametreleri sabit tutularak model parametreleri güncellenir. Yakınsama sağlanana kadar bu iki adım tekrar edilir.

Hesaplama kompleksliğine göre:

Optimizasyon yaklaşımını kullanan tek aşamalı yöntemler genellikle Yarı-Belirli Programlama (YBP (Semi-Definite Programming-SDP)), Karesel olarak Kısıtlı Karesel Programlama (KKKP (Quadratically Constrained Quadratic Programming-QCQP)), İkinci Dereceden Konik Programlama (İDKP (Second-Order Cone Programming SOCP)) problemi olarak modellenirler. Bu problemlerin çözümü kanonik destek vektör makinelerinin çözümünde kullanılan Karesel Programlama (KP (Quadratic Programming-QP)) probleminin çözümünden daha zordur.

İki aşamalı yöntemler Doğrusal Programlama (DP (Linear Programming-LP)) çözücüsünü ve iç döngüde kanonik destek vektör makineleri çözücüsünü kullanan Yarı-Sonsuz Doğrusal Programlama (YSDP (Semi-Infinite Linear Programming SILP)) problemi olarak modellenebilirler (Gönen ve Alpaydın, 2011).

4.2. DESTEK VEKTÖR REGRESYON İÇİN UYGULANAN ÇOKLU ÇEKİRDEK ALGORİTMALARI

Destek vektör makineleri yüksek boyutlu veri ve çok sayıda gözlemin bulunduğu durumlarda nonlinear regresyon problemlerinin çözümünde başarıyla uygulanmıştır. Ancak destek vektör regresyonun başarısı, sınıflandırma problemlerinde de olduğu gibi, çekirdek fonksiyonunun ve parametrelerinin uygun seçimine bağlıdır. Model seçiminde klasik yöntem en iyi çekirdek fonksiyonunun deneme-yanılma yöntemi ile seçilmesi ve ilgili parametrelerin optimum değerlerinin her bir çekirdek fonksiyonu için çapraz geçerlilik yöntemi ile belirlenmesidir. Ancak bu yöntem özellikle büyük veri setlerinde oldukça zaman alıcıdır.

Destek vektör regresyonda regresyon fonksiyonunun her iki tarafı için de doğruluk kontrolü gerektiğinden hata marjini tıpünü oluşturmak için farklı gevşek değişkenler kullanılmaktadır. Bununla birlikte destek vektör regresyon, formülasyonunda daha fazla sayıda kısıt söz konusu olduğundan destek vektör sınıflandırmadan daha kompleksdir.

Destek vektör makineleri için uygulanan çoklu çekirdek algoritmaları Çizelge 4.1'de sunulmuştur (Gönen ve Alpaydın, 2011):

Çizelge 4.1. Destek vektör makineleri için uygulanan çoklu çekirdek algoritmaları

Sıra Nu.	Kaynak	Öğrenme yöntemi	Fonksiyon formu	Amaç fonksiyonu	Eğitim yöntemi	Model
1	Lanckriet ve diğ. (2004)	Optimizasyon	Doğrusal (Konik)	Yapısal risk	Tek aşamalı (Ardışık)	KKKP + KP
2	Qiu ve Lane (2005)	Optimizasyon	Doğrusal (Konik)	Yapısal risk	Tek aşamalı (Ardışık)	KKKP + KP
3	Qiu ve Lane (2009)	Sezgisel	Doğrusal (Konveks)	-	Tek aşamalı (Ardışık)	KP
4	Bach ve diğ. (2004)	Optimizasyon	Doğrusal (Konveks)	Yapısal risk	Tek aşamalı (Eş zamanlı)	İDKP
5	Sonnenburg ve diğ. (2006a ve 2006b)	Optimizasyon	Doğrusal (Konveks)	Yapısal risk	İki aşamalı	DP + KP
6	Rakotomamonjy ve diğ. (2007 ve 2008)	Optimizasyon	Doğrusal (Konveks)	Yapısal risk	İki aşamalı	Gradient + KP
7	Varma ve Babu (2009)	Optimizasyon	Nonlineer	Yapısal risk	İki aşamalı	Gradient + KP
8	Gönen ve Alpaydın (2008 ve 2010)	Optimizasyon	Veriye bağlı	Yapısal risk	İki aşamalı	Gradient + KP

4.2.1. Çekirdek Fonksiyonlarının Kombinasyonu için Sabit Kurallar

Cristianini ve Shawe-Taylor (2000) iki geçerli çekirdek fonksiyonunun toplamlarının ve çarpımlarının da geçerli bir çekirdek fonksiyonu vereceğini göstermiştir.

$$k_{\eta}(\mathbf{x}_i, \mathbf{x}_j) = k_1(\mathbf{x}_i^1, \mathbf{x}_j^1) + k_2(\mathbf{x}_i^2, \mathbf{x}_j^2) \quad (4.6)$$

$$k_{\eta}(\mathbf{x}_i, \mathbf{x}_j) = k_1(\mathbf{x}_i^1, \mathbf{x}_j^1)k_2(\mathbf{x}_i^2, \mathbf{x}_j^2) \quad (4.7)$$

Aynı kural ikiden fazla çekirdek fonksiyonunun kombinasyonunda da geçerlidir.

$$k_{\eta}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{m=1}^P k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \quad (4.8)$$

$$k_{\eta}(\mathbf{x}_i, \mathbf{x}_j) = \prod_{m=1}^P k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \quad (4.9)$$

Sabit bir $f_{\eta}(\cdot)$ fonksiyonu kullanmak yerine parametrelili bir fonksiyon da kullanılabilir. Bu durumda fonksiyonun parametrelerinin optimize edilmesi için de bir öğrenme prosedürü uygulanır.

Çekirdek fonksiyonlarının ağırlıklı toplamları kullanılabilir. Bu durumda birleştirilmiş çekirdek fonksiyonu aşağıdaki şekilde formüle edilmektedir:

$$k_{\eta}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{m=1}^P \eta_m k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \quad (4.10)$$

Bu yaklaşım için önerilen metotlar $\{\eta_m\}_{m=1}^P$ ağırlıkları için uyguladıkları kısıtlar bakımından farklılık göstermektedir.

4.2.2. Negatif Olmayan Doğrusal Kombinasyon (Konik Kombinasyon)

Lanckriet ve diğ. (2004) çekirdek fonksiyonlarının doğrusal kombinasyonları için Yarı-Belirli Programlama yöntemini önermiştir. Konik kombinasyonda, yani kombinasyon ağırlıkları negatif değer almayacak şekilde kısıtlandığında, problemin Karesel olarak

Kısıtlı Karesel Programlama modeli olarak formüle edilebileceğini göstermiştir. Problemi çözmek için iç noktalar (interior points) yöntemini kullanmıştır.

Bu durumda birleştirilmiş çekirdek fonksiyonunun seçileceği küme aşağıdaki şekilde gösterilmektedir:

$$K_P = \left\{ \mathbf{K}: \mathbf{K} = \sum_{m=1}^P \eta_m \mathbf{K}_m, \eta_m \geq 0, \mathbf{K}_m \geq 0, iz(\mathbf{K}) \leq c \right\} \quad (4.11)$$

Karesel olarak Kısıtlı Karesel Programlama modeli aşağıdaki şekildedir:

$$\begin{aligned} \min_{\alpha, t} \quad & \frac{1}{2} ct - \sum_{i=1}^N \alpha_i \\ & iz(\mathbf{K}_m)t \geq \alpha^\top \left((yy^\top) \mathbf{K}_m^{iz} \right) \alpha \quad \forall m \\ & \sum_{i=1}^N \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq C \quad \forall i \\ & \alpha \in \mathbb{R}_+^N, t \in \mathbb{R} \end{aligned} \quad (4.12)$$

Destek vektör sınıflandırma için Lanckriet ve diğ. (2004) tarafından önerilen çoklu çekirdek öğrenme algoritmalarının regresyon problemleri için geliştirilmesi Qiu ve Lane (2005) tarafından yapılmıştır. Çekirdek fonksiyonunun seçimi problemini Yarı-Belirli Programlama ve Karesel olarak Kısıtlı Karesel Programlama olarak formüle etmişler ve çekirdek fonksiyonlarının pozitif katsayılar ile doğrusal kombinasyonu metodunu önermişlerdir.

Pozitif katsayılar sayesinde farklı çekirdek fonksiyonlarının göreceli önem dereceleri hakkında yorum yapılabilir.

Destek vektör regresyon için dual problem ve regresyon fonksiyonu aşağıdaki şekildedir:

$$\text{maks} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^+ - \alpha_i^-) (\alpha_j^+ - \alpha_j^-) k(\mathbf{x}_i, \mathbf{x}_j) - \varepsilon \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) - \sum_{i=1}^N y_i (\alpha_i^+ - \alpha_i^-)$$

$$\sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0$$

$$0 \leq \alpha_i^+, \alpha_i^- \leq C \quad (4.13)$$

$$f(\mathbf{x}) = \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) k(\mathbf{x}_i, \mathbf{x}_j) + b \quad (4.14)$$

Destek vektör regresyonun dual problemini Yarı-Belirli Programlama şeklinde formüle etmek için aşağıdaki değişkenler tanımlanmaktadır:

$$\alpha = \begin{pmatrix} \alpha^+ \\ \alpha^- \end{pmatrix} \in \mathbb{R}^{2N} \quad (4.15)$$

$$Q(\mathbf{K}) = \begin{pmatrix} \mathbf{K} & -\mathbf{K} \\ -\mathbf{K} & \mathbf{K} \end{pmatrix} \in \mathbb{R}^{2N \times 2N} \quad (4.16)$$

$$y = (y_1, \dots, y_N)^\top \quad (4.17)$$

$$h = \begin{pmatrix} -\varepsilon e + y \\ -\varepsilon e - y \end{pmatrix} \in \mathbb{R}^{2N} \quad (4.18)$$

Burada $\mathbf{K} \in \mathbb{R}^{N \times N}$ çekirdek matrisidir, $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. e birlerden oluşan bir vektördür. Ayrıca $i = 1, \dots, N$ ise $z_i = 1$; $i = N + 1, \dots, 2N$ ise $z_i = -1$ olacak şekilde $z \in \mathbb{R}^{2N}$ değişkeni tanımlanmıştır. Bu değişkenler kullanılarak Eşitlik (4.13) ile verilen dual problem aşağıdaki şekilde yazılabilir (Qiu ve Lane, 2005):

$$\max_{\alpha} 2h^\top \alpha - \alpha^\top Q(\mathbf{K}) \alpha$$

$$z^\top \alpha = 0$$

$$\alpha_i \in [0, C], \quad i = 1, \dots, 2N \quad (4.19)$$

Eğer $\eta_m \geq 0$ olacak şekilde bir kısıt eklenirse Karesel olarak Kısıtlı Karesel Programlama problemi elde edilmiş olur. \mathbf{K} çekirdek fonksiyonu seti aşağıdaki şekilde tanımlanmaktadır:

$$\mathbf{K}_P = \left\{ \mathbf{K}: \mathbf{K} = \sum_{m=1}^P \eta_m \mathbf{K}_m, \eta_m \geq 0, \mathbf{K}_m \geq 0, \text{iz}(\mathbf{K}) \leq c \right\} \quad (4.20)$$

$$\gamma = \alpha_i^+ - \alpha_i^- \quad (4.21)$$

$$\mu = \alpha_i^+ + \alpha_i^- \quad (4.22)$$

olsun. Çekirdek fonksiyonu seçimi problemi destek vektör regresyon probleminin duali ile aşağıdaki şekilde yazılabilir (Qiu ve Lane, 2005):

$$\min_{\mathbf{K}} \max_{\gamma, \mu} 2\mathbf{y}^\top \gamma - 2\epsilon \mathbf{e}^\top \mu - \gamma^\top \mathbf{K} \gamma$$

$$\mathbf{e}^\top \gamma = 0$$

$$-C \leq \gamma \leq C$$

$$0 \leq \mu \leq 2C$$

$$\text{iz}(\mathbf{K}) = c$$

$$\mathbf{K} = \sum_{m=1}^P \eta_m \mathbf{K}_m$$

$$\mathbf{K}_m \geq 0$$

$$\eta_m \geq 0 \quad (4.23)$$

Bu model aşağıdaki formulasyona eşdeğerdir:

$$\min_{\eta} \max_{\gamma, \mu} 2y^T \gamma - 2\epsilon e^T \mu - \gamma^T \mathbf{K} \gamma$$

$$e^T \gamma = 0$$

$$-C \leq \gamma \leq C$$

$$0 \leq \mu \leq 2C$$

$$\sum_{m=1}^P \eta_m \text{iz}(\mathbf{K}_m) = c$$

$$\eta_m \geq 0, \quad m = 1, \dots, P \quad (4.24)$$

Güçlü dualiteden ve $\inf(-u) = -\sup(u)$ olduğundan Eşitlik (4.24) ile verilen amaç fonksiyonunun değeri aşağıdaki şekilde hesaplanabilir (Qiu ve Lane, 2005):

$$\min_{\eta} \max_{\gamma, \mu} 2y^T \gamma - 2\epsilon e^T \mu - \gamma^T \sum_{m=1}^P \eta_m \mathbf{K}_m \gamma$$

$$= \max_{\gamma, \mu} \min_{\eta} 2y^T \gamma - 2\epsilon e^T \mu - \sum_{m=1}^P \eta_m \gamma^T \mathbf{K}_m \gamma$$

$$= \max_{\gamma, \mu} 2y^T \gamma - 2\epsilon e^T \mu + \min_{\eta} \left(- \sum_{m=1}^P \eta_m \gamma^T \mathbf{K}_m \gamma \right)$$

$$= \max_{\gamma, \mu} 2y^T \gamma - 2\epsilon e^T \mu - \max_{\eta} \sum_{m=1}^P \eta_m \gamma^T \mathbf{K}_m \gamma$$

$$= \max_{\gamma, \mu} 2y^T \gamma - 2\epsilon e^T \mu - \max_m \left(\frac{c}{\text{iz}(\mathbf{K}_m)} \gamma^T \mathbf{K}_m \gamma \right) \quad (4.25)$$

Eşitlik (4.23) ve (4.24)'deki kısıtların Eşitlik (4.25) ile verilen amaç fonksiyonuna entegre edilmesi ile çekirdek fonksiyonu seçimi problemi aşağıdaki şekilde yazılır:

$$\max_{\gamma, \mu} 2\gamma^\top \gamma - 2\epsilon e^\top \mu - ct$$

$$t \geq \frac{1}{iz(\mathbf{K}_m)} \gamma^\top \mathbf{K}_m \gamma, \quad m = 1, \dots, P \quad (4.26)$$

$$e^\top \gamma = 0$$

$$-C \leq \gamma \leq C$$

$$0 \leq \mu \leq 2C \quad (4.27)$$

Konveks programlama yazılımlarının, MOSEK gibi, kullanılabilmesi için amaç fonksiyonunun doğrusal, kısıtların ise doğrusal veya konik olması gerekmektedir. Eşitlik (4.26) ile verilen P kısıt faktörizasyon ve ilave değişkenler yardımı ile aşağıdaki şekilde standart konik forma dönüştürülebilir (Qiu ve Lane, 2005):

$$\mathbf{K}_m = L_m L_m^\top$$

$$\beta_m = L_m^\top \gamma$$

$$\gamma^\top \mathbf{K}_m \gamma = (L_m^\top \gamma)^\top L_m^\top \gamma = \beta_m^\top \beta_m$$

$$iz(\mathbf{K}_m)t = \tau^2$$

$$\tau \geq \sqrt{\beta_m^\top \beta_m} \quad (4.28)$$

$\mathbf{K}_m \geq 0$ olduğundan $\mathbf{K}_m = L_m L_m^\top$ faktörizasyonu yapılabilmektedir.

Böylece Eşitlik (4.27) ile verilen problemin standart konik formulasyonu aşağıda verildiği şekilde elde edilmektedir:

$$\begin{aligned}
& \min_{\gamma, \mu, \beta_m, v, w, s, t_m} v - 2y^\top \gamma - 2\epsilon e^\top \mu \\
& w = 1 \\
& at_1 - s = 0 \\
& e^\top \gamma = 0 \\
& -C \leq \gamma \leq C \\
& 0 \leq \mu \leq 2C \\
& t_m - t_1 = 0, \quad 2 \leq m \leq P \\
& 2vw \geq s^2 \\
& L_m^\top \gamma - \beta_m = 0, \quad m = 1, \dots, P \\
& t_m \geq \sqrt{\beta_j^\top \beta_j}, \quad m = 1, \dots, P
\end{aligned} \tag{4.29}$$

Burada; $2vw \geq s^2$ döndürülmüş konik kısıttır ve her bir \mathbf{K}_m normalleştirildiğinde $a = \sqrt{2c/iz(\mathbf{K}_m)}$ 'dir.

Çekirdek fonksiyonlarının ağırlıkları her bir çekirdek fonksiyonu tek başına kullanıldığında elde edilen performans değerlerine bakılarak belirlenebilir.

Qiu ve Lane (2009) regresyon problemlerinde çekirdek fonksiyonlarının ağırlıklarının seçimi için iki sezgisel yöntem önermişlerdir:

$$\eta_m = \frac{R_m}{\sum_{h=1}^P R_h} \quad \forall m \tag{4.30}$$

$$\eta_m = \frac{\sum_{h=1}^P M_h - M_m}{(P-1) \sum_{h=1}^P M_h} \quad \forall m \tag{4.31}$$

Burada R_m , gerçek çıktı değerleri ile \mathbf{K}_m çekirdek matrisini kullanan regresyon modeli ile elde edilen tahmin değerleri arasındaki Pearson korelasyon katsayısıdır. M_m ise \mathbf{K}_m çekirdek matrisini kullanan regresyon modelinin ortalama hata karesidir. Bu sezgisel yaklaşımlar çekirdek fonksiyonlarının konveks kombinasyonunu vermektedir.

Lewis ve diğ. (2006) ağırlıklı ve ağırlıksız olarak toplanmış çekirdek fonksiyonu kombinasyonlarının sınıflandırma problemi üzerinde performanslarını karşılaştırmıştır. Birleştirilen bütün çekirdek fonksiyonları veya veri kaynakları bilgi verici olduğunda ağırlıksız toplamların, bazı çekirdek fonksiyonları veya veri kaynakları gürültü içeriyorsa veya önemsiz ise ağırlıklı toplamların kullanılmasının ve ağırlıkların optimize edilmesinin daha faydalı olduğu sonucuna varmışlardır.

4.2.3. Konveks Kombinasyon

Çekirdek fonksiyonlarının kombinasyonu $\eta_m \in \mathbb{R}_+$ ve $\sum_{m=1}^P \eta_m = 1$ olmak üzere ağırlıklı ortalamaları da olabilir. Joachims ve diğ. (2001) eğer iki çekirdek fonksiyonu yaklaşık olarak aynı performansı gösteriyor ve farklı veri örneklerini destek vektör olarak kullanıyorsa iki çekirdek fonksiyonunu birleştirmenin faydalı olacağını göstermiştir. Bu durumda iki çekirdek fonksiyonu birbirini tamamlayıcı özelliktedir ($\eta_2 = 1 - \eta_1$).

Bach ve diğ. (2004), Karesel olarak Kısıtlı Karesel Programlama modelinin dualini İkinci Dereceden Konik Programlama problemi olarak formüle etmiştir. Sıralı Minimal Optimizasyon tekniklerinin uygulanabileceği bir formulasyon elde etmek için Moreau-Yosida düzleştirme (regularization) tekniğinden faydalanmıştır.

Bach ve diğ. (2004) primal formulasyonu aşağıdaki şekilde modifiye etmiştir:

$$\begin{aligned} \min_{\mathbf{w}_m, \xi, b} \frac{1}{2} \left(\sum_{m=1}^P d_m \|\mathbf{w}_m\|_2 \right)^2 + C \sum_{i=1}^N \xi_i \\ y_i \left(\sum_{m=1}^P \langle \mathbf{w}_m, \Phi_m(\mathbf{x}_i^m) \rangle + b \right) \geq 1 - \xi_i \quad \forall i \\ \mathbf{w}_m \in \mathbb{R}^{S_m}, \quad \xi \in \mathbb{R}_+^N, \quad b \in \mathbb{R} \end{aligned} \quad (4.32)$$

Eğer bu optimizasyon problemi İkinci Dereceden Konik Programlama modeli olarak düşünülürse dual formulasyon aşağıdaki şekilde elde edilir:

$$\begin{aligned}
& \min_{\gamma, \alpha} \frac{1}{2} \gamma^2 - \sum_{i=1}^N \alpha_i \\
& \gamma^2 d_m^2 \geq \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \quad \forall m \\
& \sum_{i=1}^N \alpha_i y_i = 0 \\
& 0 \leq \alpha_i \leq C \quad \forall i \\
& \gamma \in \mathbb{R}, \alpha \in \mathbb{R}_+^N
\end{aligned} \tag{4.33}$$

Ağırlıklar aşağıdaki kısıtı sağlamaktadır:

$$\sum_{m=1}^P d_m^2 \eta_m = 1 \tag{4.34}$$

Dual problem $d_m = \sqrt{iz(\mathbf{K}_m)/c}$ alındığında Lanckriet ve diğ. (2004) tarafından önerilen Karesel olarak Kısıtlı Karesel Programlama modeline eş olmaktadır. İkinci Dereceden Konik Programlama modelinin avantajı Bach ve diğ. (2004) tarafından primal optimizasyon modelinin amaç fonksiyonuna Moreau-Yosida düzleştirme teriminin, $1/2 \sum_{m=1}^P \alpha_m^2 \|\mathbf{w}_m\|_2^2$, eklenmesiyle ilgili dual modelin oluşturularak Sıralı Minimal Optimizasyon tekniğinin kullanılmasının önerilmesidir.

Sonnenburg ve diğ. (2006a ve 2006b) modeli Yarı-Sonsuz Doğrusal Programlama problemi olarak formüle etmiş ve problemin çözümünde sütun türetme (column generation) tekniğini kullanmıştır. Önerilen algoritmanın yüz binlerce örnek üzerinde ve yüzlerce çekirdek kombinasyonu için çözülebildiğini göstermiştir. Bununla birlikte otomatik model seçimi ve sonuçların yorumlanması konularında da katkı sağladığını göstermiştir. Önerilen yöntem destek vektör regresyon için genelleştirilmiştir. Çoklu çekirdek öğrenme algoritmaları SHOGUN machine learning toolbox (Sonnenburg ve diğ., 2010)'ına entegre edilmiştir.

Destek vektör makinelerinin diğer tekniklere göre dezavantajı elde edilen karar fonksiyonunun yorumlanmasındaki güçlüklerdir. Bu nedenle karar fonksiyonunun problem hakkında bilgi edinmek için kullanılması zor olmaktadır. Çoklu çekirdek öğrenme formülasyonları mevcut çekirdek tabanlı öğrenme algoritmalarının bu önemli eksik özelliği hakkında faydalı çıkarımlar sağlamaktadır.

Sonnenburg ve diğ. (2006a ve 2006b), çekirdek fonksiyonlarının konveks kombinasyonlarını ele almıştır.

$$k_{\eta}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{m=1}^P \eta_m k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \quad (4.35)$$

$$\eta_m \geq 0 \quad \text{ve} \quad \sum_{m=1}^P \eta_m = 1 \quad (4.36)$$

Sonnenburg ve diğ. (2006a ve 2006b), Bach ve diğ. (2004) tarafından önerilen Karesel olarak Kısıtlı Karesel Programlama problemini aşağıdaki şekilde yeniden yazmıştır:

$$\min_{\gamma, \alpha} \gamma$$

$$\sum_{i=1}^N \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq C$$

$$\gamma \geq \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) - \sum_{i=1}^N \alpha_i \quad \forall m$$

$$\gamma \in \mathbb{R}, \alpha \in \mathbb{R}_+^N \quad (4.37)$$

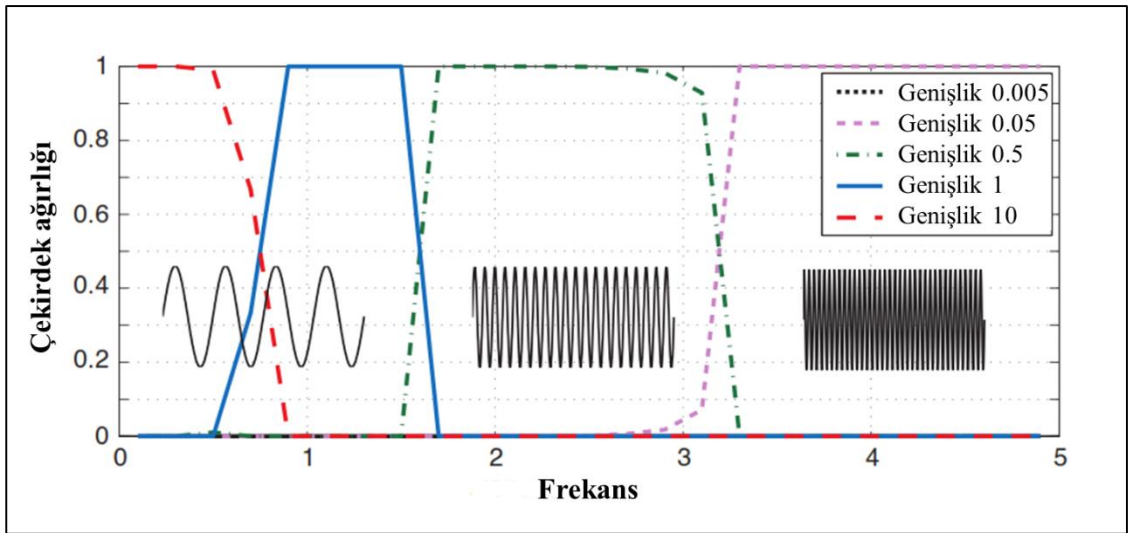
$$S_m(\alpha) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) - \sum_{i=1}^N \alpha_i \quad (4.38)$$

olmak üzere modeli Yarı-Belirli Doğrusal Programlama problemine dönüştürmüştür.

$$\begin{aligned}
& \max_{\theta, \eta} \theta \\
& \sum_{m=1}^P \eta_m = 1 \\
& \sum_{m=1}^P \eta_m S_m(\alpha) \geq \theta \quad \forall \alpha \in \{\alpha: \alpha \in \mathbb{R}^N, \quad \alpha^\top y = 0, \quad 0 \leq \alpha \leq C\} \\
& \theta \in \mathbb{R}, \quad \eta \in \mathbb{R}_+^P
\end{aligned} \tag{4.39}$$

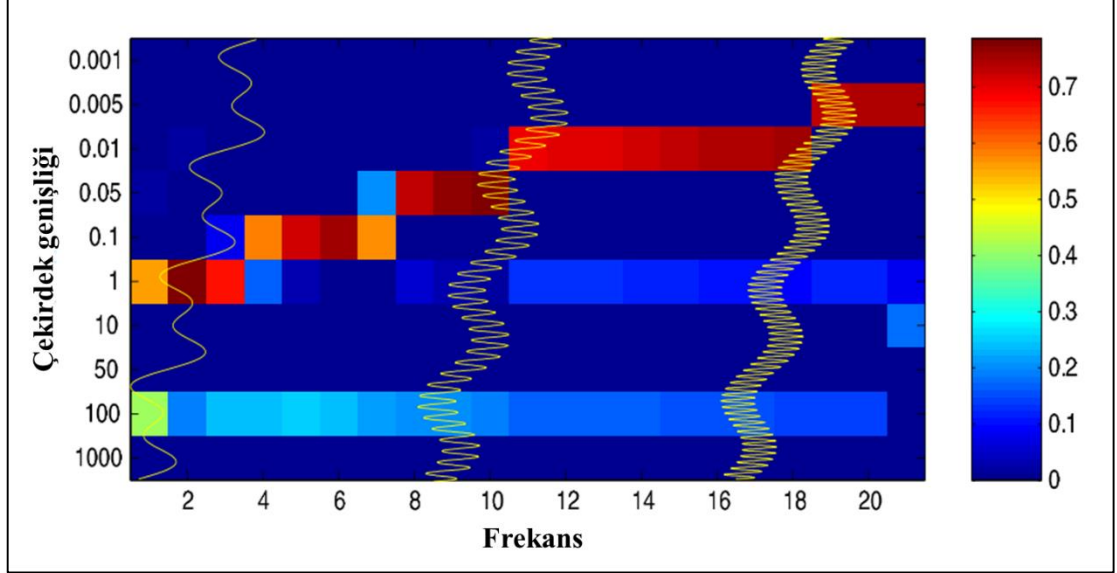
Sonnenburg ve diğ. (2006a ve 2006b) problemin çözümünde sütun türetme tekniğini kullanmıştır, Doğrusal Programlama çözücü (solver) ve iç döngüde kanonik destek vektör makineleri çözücüsünü kullanmıştır.

Sonnenburg ve diğ. (2006a ve 2006b) çoklu çekirdek destek vektör regresyon yöntemini farklı genişliklere sahip Gaussian çekirdek fonksiyonlarını kullanarak sinüs fonksiyonunu öğrenmek için uygulamıştır. Bu maksatla artan frekanslara sahip sinüs dalgaları veri setleri oluşturulmuştur. Şekil 4.1’de örnek olarak üç sinüs dalgası sunulmuştur. Frekans soldan sağa doğru artmaktadır. Her frekans için farklı çekirdek genişlikleri ile hesaplanan ağırlıklar gösterilmiştir. Şekil 4.1’den anlaşılacağı üzere çoklu çekirdek destek vektör regresyon algoritması regresyon problemine en uygun olacak şekilde çekirdek ağırlıklarını değiştirmektedir.



Şekil 4.1. Sinüs dalgası problemi (Sonnenburg ve diğ., 2006a ve 2006b)

Diğer uygulamada ise birisi yüksek frekansa, diğeri düşük frekansa sahip olan iki sinüs dalgası doğrusal fonksiyon kullanılarak birleştirilmiştir, $f(x) = \sin(ax) + \sin(bx) + cx$. Bununla birlikte, yüksek frekansa sahip olan sinüs dalgasının frekansı arttırılmıştır, yani b ve c sabit tutulurken a değiştirilmiştir. Çoklu çekirdek algoritmaları ile elde edilen ağırlıklar farklı çekirdek fonksiyonlarının kombinasyonunu verecektir. Farklı genişliklere sahip 10 adet Gaussian çekirdek fonksiyonu kullanılmıştır. Çoklu çekirdek destek vektör regresyon yönteminin uygulanması sonucunda elde edilen ağırlıklar Şekil 4.2’de sütun şeklinde sunulmuştur. Seçilen en büyük genişlik (100), doğrusal bileşeni modellemektedir. Orta genişlik (1), düşük frekanslı sinüs dalgasına karşılık gelmektedir. Yüksek frekanslı sinüs dalgasının frekansı soldan sağa doğru arttırılmıştır. Çoklu çekirdek algoritmaları frekans arttıkça Gaussian çekirdek fonksiyonunun genişliğini azaltarak düşük ve yüksek genişlikli çekirdeklerin uygun bir kombinasyonunu bulmaktadır. Ayrıca çoklu çekirdek algoritmaları sadece gerekli olan çekirdekleri kombinasyona dâhil etmektedir. Şekil 4.2’de görüleceği üzere çoğu çekirdek ağırlığı mavi ile gösterilmiştir, yani sıfıra eşittir.



Şekil 4.2. Sinüs fonksiyonlarının kombinasyonu (Sonnenburg ve diğ., 2006a ve 2006b) Rakotomamonjy ve diğ. (2007 ve 2008) çoklu çekirdek öğrenme için farklı bir primal problem önermiş ve modeli çözmek için gradient yöntemini kullanmıştır. Önerilen

yöntem SimpleMKL (Simple Multiple Kernel Learning - Basit Çoklu Çekirdek Öğrenme) olarak adlandırılmış ve destek vektör regresyon için de genişletilmiştir.

Rakotomamonjy ve diğ. (2007 ve 2008) tarafından önerilen primal problem aşağıdaki şekildedir:

$$\begin{aligned}
\min_{\mathbf{w}_m, \xi, b, \eta} \quad & \frac{1}{2} \sum_{m=1}^P \frac{1}{\eta_m} \|\mathbf{w}_m\|_2^2 + C \sum_{i=1}^N \xi_i \\
y_i \left(\sum_{m=1}^P \langle \mathbf{w}_m, \Phi_m(\mathbf{x}_i^m) \rangle + b \right) & \geq 1 - \xi_i \quad \forall i \\
\sum_{m=1}^P \eta_m & = 1 \\
\mathbf{w}_m \in \mathbb{R}^{S_m}, \quad \xi \in \mathbb{R}_+^N, \quad b \in \mathbb{R}, \quad \eta \in \mathbb{R}_+^P & \quad (4.40)
\end{aligned}$$

Verilen η için $J(\eta)$ aşağıdaki şekilde tanımlanmaktadır:

$$\begin{aligned}
\min_{\mathbf{w}_m, \xi, b} \quad & J(\eta) = \frac{1}{2} \sum_{m=1}^P \frac{1}{\eta_m} \|\mathbf{w}_m\|_2^2 + C \sum_{i=1}^N \xi_i \\
y_i \left(\sum_{m=1}^P \langle \mathbf{w}_m, \Phi_m(\mathbf{x}_i^m) \rangle + b \right) & \geq 1 - \xi_i \quad \forall i \\
\mathbf{w}_m \in \mathbb{R}^{S_m}, \quad \xi \in \mathbb{R}_+^N, \quad b \in \mathbb{R} & \quad (4.41)
\end{aligned}$$

Güçlü dualiteden dolayı $J(\eta)$ dual formülasyondan faydalanılarak hesaplanabilir.

$$\begin{aligned}
\max_{\alpha} \quad & J(\eta) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j k_{\eta}(\mathbf{x}_i, \mathbf{x}_j) \\
\sum_{i=1}^N \alpha_i y_i & = 0 \\
0 \leq \alpha_i \leq C \quad & \forall i \\
\alpha \in \mathbb{R}_+^N & \quad (4.42)
\end{aligned}$$

Primal problem aşağıdaki kısıtlı optimizasyon problemi halini almaktadır:

$$\begin{aligned}
 & \min_{\eta} J(\eta) \\
 & \sum_{m=1}^P \eta_m = 1 \\
 & \eta \in \mathbb{R}_+^P
 \end{aligned} \tag{4.43}$$

Önerilen yöntem SimpleMKL olarak adlandırılmaktadır ve iki temel aşamadan oluşmaktadır. Birincisi verilen η için kanonik destek vektör makineleri optimizasyon probleminin çözülmesi, ikincisi ise ilk adımda bulunan α ile hesaplanan aşağıdaki gradient kullanılarak η ağırlıklarının güncellenmesidir.

$$\frac{\partial J(\eta)}{\partial \eta_m} = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j k_{\eta}(\mathbf{x}_i^m, \mathbf{x}_j^m) \quad \forall m \tag{4.44}$$

Gradient güncelleme prosedürü çekirdek ağırlıklarının negatif olmaması ve normleştirilmesine dayanmaktadır. Önerilen algoritma Sonnenburg ve diğ. (2006a ve 2006b) tarafından kullanılan algoritmaya oldukça benzerdir. Her iki algoritma da her bir adımda α 'nın hesaplanması için kanonik destek vektör makineleri çözücüsünü kullanmaktadır. Aralarındaki fark η ağırlıklarının güncellenmesi için kullanılan prosedürdür: birisi gradient tekniğini kullanırken diğeri Doğrusal Programlama çözücüsünü kullanmaktadır.

4.2.4. Çekirdek Fonksiyonlarının Nonlinear Kombinasyonu

Varma ve Babu (2009) çoklu çekirdek öğrenme algoritmaları için geliştirilmiş bir formulasyon önermiştir. Bu formulasyonda amaç fonksiyonu iki düzeltme terimi (regularization term) ve bir kayıp fonksiyonu içermektedir. Önerilen formulasyon Genelleştirilmiş Çoklu Çekirdek Öğrenme (Generalized Multiple Kernel Learning - GMKL) olarak adlandırılmıştır.

Önerilen formulasyon hem hiperdüzlem ağırlıklarını hem de çekirdek kombinasyon ağırlıklarını düzenlemeye çalışmaktadır. Kayıp fonksiyonu klasik kayıp

fonksiyonlarından birisi olabilir (sınıflandırma için hinge kayıp fonksiyonu ya da regresyon için ε -duyarsız kayıp fonksiyonu gibi). $r(\cdot)$ düzleştirme (regularization) fonksiyonu olmak üzere önerilen primal formülasyon aşağıdaki şekildedir:

$$\min_{\mathbf{w}_\eta, \xi, b, \eta} \frac{1}{2} \|\mathbf{w}_\eta\|_2^2 + C \sum_{i=1}^N \xi_i + r(\eta)$$

$$y_i (\langle \mathbf{w}_\eta, \Phi_\eta(\mathbf{x}_i) \rangle + b) \geq 1 - \xi_i \quad \forall i$$

$$\mathbf{w}_\eta \in \mathbb{R}^{S_\eta}, \quad \xi \in \mathbb{R}_+^N, \quad b \in \mathbb{R}, \quad \eta \in \mathbb{R}_+^P \quad (4.45)$$

Burada $\Phi_\eta(\cdot)$, $k_\eta = (\cdot, \cdot)$ birleştirilmiş çekirdek fonksiyonunun ait olduğu özellik uzayına karşılık gelmektedir. \mathbf{w}_η ise $\Phi_\eta(\cdot)$ 'ye atanan ağırlık katsayıları vektörüdür. Bu problem SimpleMKL'den farklı olarak konveks değildir, fakat çözüm stratejisi aynıdır. Verilen η için primal formülasyonun amaç fonksiyonu değeri hedef fonksiyon olarak kullanılmaktadır.

$$\min_{\mathbf{w}_\eta, \xi, b} J(\eta) = \frac{1}{2} \|\mathbf{w}_\eta\|_2^2 + C \sum_{i=1}^N \xi_i + r(\eta)$$

$$y_i (\langle \mathbf{w}_\eta, \Phi_\eta(\mathbf{x}_i) \rangle + b) \geq 1 - \xi_i \quad \forall i$$

$$\mathbf{w}_\eta \in \mathbb{R}^{S_\eta}, \quad \xi \in \mathbb{R}_+^N, \quad b \in \mathbb{R}, \quad (4.46)$$

Gradient adımı için aşağıda verilen dual formülasyon kullanılmaktadır (Varma ve Babu, 2009):

$$\max_{\alpha} J(\eta) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j k_\eta(\mathbf{x}_i, \mathbf{x}_j) + r(\eta)$$

$$\sum_{i=1}^N \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq C \quad \forall i$$

$$\alpha \in \mathbb{R}_+^N \quad (4.47)$$

Çekirdek ağırlıklarına göre gradient aşağıdaki şekilde hesaplanmaktadır (Varma ve Babu, 2009):

$$\frac{\partial J(\eta)}{\partial \eta_m} = \frac{\partial r(\eta)}{\partial \eta_m} - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \frac{k_\eta(\mathbf{x}_i, \mathbf{x}_j)}{\partial \eta_m} \quad \forall m \quad (4.48)$$

4.2.5. Veriye Bağlı Çoklu Öğrenme Algoritmaları

Gönen (2010) makine öğrenme için veriye bağlı (yerel) çoklu çekirdek yöntemini önermiştir. Bu yöntem çekirdek-tabanlı öğrenme algoritması ve çekirdek fonksiyonlarına yerel ağırlıklar atamak için kullanılan parametrik geçit (gating) modeli (softmax, sigmoid ve Gaussian) olmak üzere iki bileşenden oluşmaktadır. Bu iki bileşen birbirini izleyen iki adımlı optimizasyon algoritması ile eğitilmektedir.

Ağırlıklı veya ağırlıksız toplamlar çekirdek fonksiyonlarına tüm girdi uzayı için aynı ağırlıkları atamaktadır. Çekirdek fonksiyonuna girdi uzayının farklı bölgeleri için farklı ağırlıklar atanması daha iyi sonuç verebilecektir. Eğer verinin temelini oluşturan yerel yapılar mevcut ise farklı bölgeler için farklı benzerlik ölçüleri daha uygun olabilir. Gönen ve Alpaydın (2008) geçit (gating) fonksiyonu kullanarak girdi uzayını bölgelere ayırmayı ve çekirdek fonksiyonlarına kombinasyon ağırlıklarını veriye bağlı olarak atamayı önermiştir.

Gönen ve Alpaydın (2010) yerel çekirdek kombinasyonu düşüncesini destek vektör regresyon yöntemi için genişletmiştir. Karar fonksiyonu aşağıdaki şekilde yazılmaktadır:

$$f(\mathbf{x}) = \sum_{m=1}^P \eta_m(\mathbf{x}|V) \langle w_m, \Phi_m(\mathbf{x}^m) \rangle + b \quad (4.49)$$

Modifiye edilmiş primal optimizasyon problemi ise aşağıdaki şekildedir (Gönen ve Alpaydın, 2010):

$$\begin{aligned}
& \min_{\mathbf{w}_m, \xi^+, \xi^-, V, b} \frac{1}{2} \sum_{m=1}^P \|\mathbf{w}_m\|_2^2 + C \sum_{i=1}^N (\xi_i^+ + \xi_i^-) \\
& y_i - \sum_{m=1}^P \eta_m(\mathbf{x}_i|V) \langle \mathbf{w}_m, \Phi_m(\mathbf{x}_i^m) \rangle - b \leq \varepsilon + \xi_i^+ \quad \forall i \\
& \sum_{m=1}^P \eta_m(\mathbf{x}_i|V) \langle \mathbf{w}_m, \Phi_m(\mathbf{x}_i^m) \rangle + b - y_i \leq \varepsilon + \xi_i^- \quad \forall i \\
& \mathbf{w}_m \in \mathbb{R}^{S_m}, \quad \xi^+, \xi^- \in \mathbb{R}_+^N, \quad V \in \mathbb{R}^{P \times (D_G + 1)}, \quad b \in \mathbb{R}
\end{aligned} \tag{4.50}$$

Verilen V için dual formülasyon aşağıdaki şekilde yazılmaktadır:

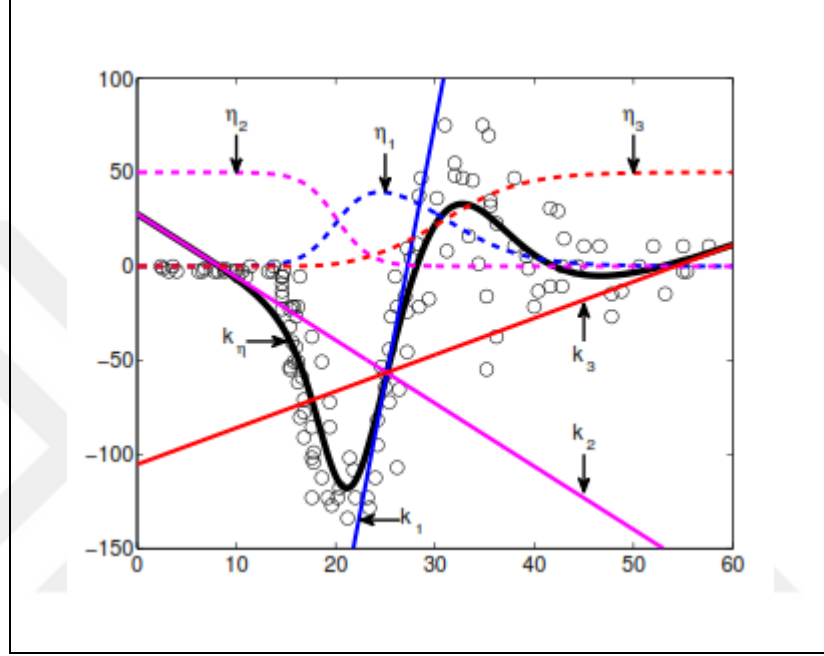
$$\begin{aligned}
\max_{\alpha^+, \alpha^-} J(V) &= \sum_{i=1}^N y_i (\alpha_i^+ - \alpha_i^-) - \varepsilon \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) \\
&\quad - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^+ - \alpha_i^-) (\alpha_j^+ - \alpha_j^-) k_\eta(\mathbf{x}_i, \mathbf{x}_j) \\
&\quad \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0 \\
&\quad 0 \leq \alpha_i^+, \alpha_i^- \leq C \quad \forall i \\
&\quad \alpha^+, \alpha^- \in \mathbb{R}_+^N
\end{aligned} \tag{4.51}$$

Elde edilen karar fonksiyonu aşağıdaki şekildedir:

$$f(\mathbf{x}) = \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) k_\eta(\mathbf{x}_i, \mathbf{x}_j) + b \tag{4.52}$$

Gönen (2010) yerel çoklu çekirdek algoritmasını softmax geçit modeli ile üç doğrusal çekirdek fonksiyonu üzerinde uygulamıştır. Şekil 4.3'te 10-katlı çapraz geçerlilik sonucunda elde edilen ortalama global ve yerel tahmin ediciler sunulmuştur. Her bölgede

doğrusal çekirdek fonksiyonu kullanılarak elde edilen üç yerel model ile parçalı doğrusal bir regresyon fonksiyonu oluşturulmuştur. Bu modeller Şekil 4.3'te kesikli çizgiler ile gösterilen softmax geçit modeli ile birleştirilmiştir. Softmax geçit modeli girdi uzayını çekirdek fonksiyonları arasında bölmektedir.



Şekil 4.3. Yerel çoklu çekirdek algoritmaları (Gönen, 2010)

5. DEĞİŞKEN SEÇİMİ

Değişken ve özellik seçimi özellikle on binlerce değişken ve/veya özelliğin mevcut olduğu veri setleri üzerinde yapılan araştırmalar için önemli bir çalışma alanı teşkil etmektedir.

Değişken seçiminin amaçları şu şekilde sıralanabilir:

- i.* Boyut indirgeme ve modelin tahmin doğruluğunun geliştirilmesi,
- ii.* Eğitim süresinin kısaltılması, veri depolama ihtiyacının azaltılması, daha hızlı, uygulanması kolay ve maliyet-etkin tahmin ediciler elde edilmesi,
- iii.* Verinin görselleştirilebilmesine olanak sağlanması, verinin anlaşılabilirliğinin ve sonuçların yorumlanabilirliğinin artırılması.

Makine öğrenme problemlerinde değişken ve özellik seçimi üzerine kapsamlı bir çalışma Guyon ve Elisseeff (2003) tarafından yapılmıştır.

Değişken seçimi problemi için önerilen iki temel yaklaşım mevcuttur. Bunların ilkinde yanıt değişkeni ile ilgili olan tüm değişkenlerin bulunması veya derecelendirilmesi, ikincisinde ise iyi bir tahmin edici için faydalı olan değişken alt kümesinin seçimine odaklanılmaktadır. En ilgili olan değişkenlerin seçilmesi ile, özellikle de modele katkısı olmayan değişkenlerin mevcut olması durumunda, ideal olmayan bir tahmin edicinin elde edilmesi söz konusu olabilmektedir. Buna karşılık en faydalı değişkenlerden oluşan bir alt kümenin seçimi çoğu gereksiz değişkeni model dışında bırakabileceği gibi bazı ilgili değişkenler de model dışında kalabilecektir. İlgililik ve faydalılık kavramlarının karşılaştırılmasına ilişkin detaylı bir inceleme Kohavi ve John (1997) tarafından sunulmuştur.

Bu bölümde ilk olarak değişkenlerin korelasyon katsayılarına göre derecelendirilmesine dayanan değişken seçimi için filtreleme yöntemleri ele alınacaktır. Daha sonra alt küme

seçimi yöntemleri incelenecektir. Bunlar, sarmalama (wrapper) ve gömülü (embedded) yöntemleri kapsamaktadır.

Sarmalama yöntemlerinde değişkenler tahmin edici için faydalılıklarına göre değerlendirilmektedir. Gömülü yöntemler, aynı amaca hizmet etmekle birlikte, uyum iyiliği terimi ve büyük değişken sayıları için ceza katsayısı ile iki kısımlı amaç fonksiyonunu doğrudan optimize ederek daha etkin bir uygulama sunmaktadır.

5.1. DEĞİŞKEN DERECELENDİRME

Çoğu değişken seçimi algoritmasında değişken derecelendirme, uygulanmasının basit olması ve başarılı sonuçlar vermesi sebebiyle, temel veya yardımcı yöntem olarak kullanılmaktadır. Korelasyon yöntemleri bu kategoride ele alınmaktadır.

Veri seti $\{(x_i, y_i)\}_{i=1}^N$ şeklinde verilmiş olsun. Burada $x_i \in \mathbb{R}^D$, D boyutlu açıklayıcı değişkenler ve $y_i \in R$ yanıt değişkenidir. Değişken derecelendirme x_{ij} ve y_i değerlerinden hesaplanan $S(j)$ skor fonksiyonu kullanılarak yapılmaktadır. Yüksek skor değerine sahip olan değişkenlerin önemli olduğu kabul edilmekte ve değişkenler $S(j)$ değerleri büyükten küçüğe olacak şekilde sıralanmaktadır. Değişken derecelendirmesinin tahmin edici model oluşturulmasında kullanılması maksadıyla iç içe geçmiş alt kümeler (nested subsets) tanımlanmakta ve değişkenler önem derecelerine göre sırasıyla modele dâhil edilmektedir.

Değişken derecelendirme bir filtreleme yöntemi olarak kabul edilmekte, ön işleme (preprocessing) adımı olarak ve seçilen tahmin ediciden bağımsız bir şekilde uygulanmaktadır. Bu yöntem sadece D adet skorun hesaplanmasını ve skorların sıralanmasını gerektirdiği için hesaplama açısından etkin olduğundan tercih edilebilmektedir.

Doğrusal regresyonda korelasyon katsayısının karesi açıklama katsayısı olarak adlandırılmakta ve yanıt değişkeninin açıklayıcı değişkenler tarafından hangi oranda açıklanabildiğinin göstergesi olarak kullanılmaktadır. Bu sebeple açıklama katsayısı değişken derecelendirilmesinde kullanıldığında doğrusal uyumun iyiliğine bağlı bir derecelendirme elde edilmiş olmaktadır.

Değişken derecelendirme yöntemlerinin sağladığı faydalar sınırlıdır. Örneğin yüksek ilişkili iki değişkenden birisi gereksiz olabileceği gibi tamamlayıcı bir role de sahip olabilmektedir. İki değişken tek başlarına önemsiz olmalarına rağmen modelde beraber kullanıldıklarında tahmin edicinin performansının iyileştirilmesine önemli fayda sağlayabilmektedirler.

Bu örnekler değişkenlerin bireysel tahmin edici güçlerine göre derecelendirilmesinden ziyade beraber iyi tahmin edici gücüne sahip değişken alt kümesi seçiminin önemini göstermektedir. Takip eden kısımda alt küme seçim yöntemleri incelenecektir.

5.2. DEĞİŞKEN ALT KÜME SEÇİMİ

Alt küme seçimi yöntemlerinde değişkenlerin bireysel tahmin güçlerine göre sıralanmasından ziyade birlikte iyi tahmin gücüne sahip olan değişken alt kümesi seçilmektedir. Bu kısımda alt küme seçimi problemi ele alınacak ve çözüm için önerilen yöntemler incelenecektir.

Alt küme seçimi yöntemleri filtreleme (filter), sarmalama (wrapper) ve gömülü (embedded) yöntemler olarak sınıflandırılmaktadır. Filtre yöntemleri kullanılarak uygulanacak olan regresyon modelinden bağımsız bir şekilde önışleme adımı olarak değişken seçimi yapılmaktadır. Sarmalama yöntemlerinde değişken alt kümeleri tahmin edici güçlerine göre değerlendirilmektedir. Gömülü yöntemlerde ise değişken seçimi eğitim aşamasında yapılmakta olup, bu yöntemler genellikle seçilen makine öğrenme yöntemine özgüdür.

5.2.1. Sarmalama ve Gömülü Yöntemler

Sarmalama yöntemleri Kohavi ve John (1997) tarafından önerilmiştir ve seçilen makine öğrenme yöntemine bağlı olmaksızın değişken seçimi problemine basit ve etkin bir çözüm sunmaktadır. Genel olarak değişken alt kümelerinin göreceli önemini değerlendirmek için regresyon modelinin tahmin performansından faydalanılır. Burada göz önünde bulundurulması gereken üç konu söz konusudur: bütün olası alt kümeler uzayının nasıl araştırılacağı, araştırmanın yönünün belirlenmesi ve durdurulması için regresyon

modelinin performansının nasıl değerlendirileceği ve hangi makine öğrenme yönteminin kullanılacağı.

Performans değerlendirmesi genel olarak geçerlilik seti veya çapraz-geçerlilik yöntemi kullanılarak yapılmaktadır.

Değişken sayısının büyük olduğu durumlarda bütün olası alt kümelerin incelenmesi oldukça zaman alıcı bir aşama haline gelmektedir. Değişken alt kümelerinin araştırılmasında kullanılan yöntemlerden birisi adımsal arama stratejileridir ve bu yöntem ikiye ayrılır: ileri doğru seçim ve geriye doğru eleme. İleri doğru seçim yönteminde değişkenler aşamalı olarak modele dâhil edilmekte, geriye doğru eleme yönteminde ise bütün değişkenleri kapsayan küme ile başlanmakta ve en az önemli olan değişkenler aşamalı olarak elenmektedir. Her iki yöntem de iç içe geçmiş değişken alt kümelerini vermektedir.

Gömülü yöntemler değişken seçimini eğitim aşamasının bir parçası olarak dâhil ederek daha etkin çözümler sunmaktadır. Eğitim verisini eğitim ve geçerlilik setlerine ayırmaya gerek duyulmadığından mevcut verinin daha etkin bir şekilde kullanılmasına olanak tanır. İncelenen her bir değişken alt kümesi için tahmin edicinin en baştan uygulanması gerekmediği için sonuca daha hızlı ulaşılmasını sağlar. Takip eden kısımlarda gömülü yöntemlerin iki örneğine yer verilecektir.

5.2.2. İç İçe Geçmiş Alt Kümeler Yöntemi

Bazı gömülü yöntemlerde değişken alt kümeleri uzayında izlenecek yön amaç fonksiyonunda meydana gelen değişiklikler değerlendirilerek belirlenmektedir. Adımsal arama stratejileri (ileri doğru seçim ya da geriye doğru eleme) ile birlikte kullanıldığında değişkenlerin iç içe geçmiş kümelerini vermektedir.

İçinde bulunulan algoritma adımında seçilen değişkenlerin sayısı s ve bu değişken seti kullanıldığında uygulanan makine öğrenme yönteminin elde edilen amaç fonksiyonun değeri $J(s)$ olsun. Amaç fonksiyonundaki değişikliği hesaplamak için önerilen üç yöntem mevcuttur:

i. Sonlu (Ölçülebilir) farkın hesaplanması: Modele eklenmek veya modelden elenmek üzere aday her bir değişken için $J(s)$ ve $J(s + 1)$ ya da $J(s - 1)$ arasındaki fark hesaplanır.

ii. Maliyet fonksiyonunun kuadratik yaklaşımı: Bu yöntem LeCun ve diğ. (1989) tarafından sinir ağları için önerilmiştir ve değişkenlerin geriye doğru elenmesi için kullanılmaktadır. J 'nin ikinci dereceden Taylor açılımı hesaplanmaktadır. Optimum J değerinde birinci terim göz ardı edilebilir, bu da j değişkeni için $DJ_j = (1/2) \frac{\partial^2 J}{\partial \mathbf{w}_j^2} (D\mathbf{w}_j)^2$ varyansını verir. Ağırlıklardaki değişim $D\mathbf{w}_j = \mathbf{w}_j$, j değişkeninin modelden çıkarılmasına karşılık gelmektedir.

iii. Amaç fonksiyonunun duyarlılığının hesaplanması: Bu yöntemde ise J 'nin \mathbf{x}_j veya \mathbf{w}_j 'ye göre türevinin mutlak değeri veya karesi kullanılmaktadır.

Bazı modellerde kesin fark miktarı etkin olarak hesaplanabilmekte ve birinci yöntemin kullanılması mümkün olmaktadır. Örneğin doğrusal en küçük kareler regresyon modelinde ileri doğru seçim yöntemi her adımda hata kareler ortalamasını en fazla azaltan değişken modele dâhil edilerek uygulanmaktadır.

Çekirdek yöntemlerinde ise fark yaklaşık olarak hesaplanmaktadır. Daha önceki bölümlerde açıklandığı üzere çekirdek fonksiyonları aşağıdaki şekilde ifade edilirler (Schölkopf ve Smola, 2002):

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i k(\mathbf{x}, \mathbf{x}_i) \quad (5.1)$$

Burada k çekirdek fonksiyonu \mathbf{x} ile \mathbf{x}_i arasındaki benzerliği ölçmektedir. $J(s)$ 'nin varyansı α_i değerleri sabit tutularak hesaplanmaktadır. Bu prosedür ilk olarak destek vektör makineleri için Guyon ve diğ. (2002) tarafından önerilmiş, Rakotomamonjy (2003) ve Weston ve diğ. (2003) tarafından kullanılmıştır.

İkinci yöntem Rivals ve Personnaz (2003) tarafından önerilmiştir. Doğrusal tahmin edicinin söz konusu olduğu, $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$, durumların uygulaması daha basittir. Bu yöntemde kriter olarak \mathbf{w}_j ağırlıkların büyüklükleri yerine DJ_j değerlerinin kullanılması

önerilmektedir. Ancak karesel bir amaç fonksiyonu ile eğitilen doğrusal tahmin ediciler için bu iki kriter birbirine eşit olmaktadır. Örneğin $J = \sum_{i=1}^N (\mathbf{w} \cdot \mathbf{x}_i + b - y_i)^2$ eşitliğini kullanan doğrusal en küçük kareler modeli için ve $J = (1/2) \|\mathbf{w}\|^2$ eşitliğini minimize eden doğrusal destek vektör makineleri ya da optimum marjın sınıflandırıcıları için bu durum geçerlidir (Vapnik, 1982).

Rakotomamonjy (2003) doğrusal destek vektör makineleri için birinci ve üçüncü yöntemlerin aynı şekilde her adımda eleme için en küçük $|\mathbf{w}_j|$ değerine sahip olan değişkenleri seçtiklerini göstermiştir.

Amaç fonksiyonunun \mathbf{w}_j 'deki değişikliğe karşı duyarlılığı, üçüncü yöntem, Perkins ve diğ. (2003) tarafından ileri doğru seçme prosedüründe kullanılmış ve çapraz-entropi amaç fonksiyonu ile doğrusal modele uygulanmıştır. Önerilen modelde kriter, $\rho_i = y_i f(\mathbf{x}_i)$ olmak üzere, $\frac{\partial J}{\partial \mathbf{w}_j} = \sum_{i=1}^N \frac{\partial J}{\partial \rho_i} \frac{\partial \rho_i}{\partial \mathbf{w}_j}$ ifadesinin mutlak değeridir. Doğrusal model durumunda, $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$, kullanılan kriter geometrik olarak amaç fonksiyonunun marjın değerlerine ve $\left[\frac{\partial \rho_i}{\partial \mathbf{w}_j} = \mathbf{x}_{ij} y_i \right]_{i=1, \dots, N}$ vektörüne göre gradienti olarak yorumlanmaktadır.

Çapraz-entropi kayıp fonksiyonu için $\frac{\partial J}{\partial \rho_i} = \frac{1}{1+e^{\rho_i}}$ 'dir.

Amaç fonksiyonunun her seferinde birini dışarıda bırakma (leave one out) çapraz geçerlilik hatası ile değiştirilmesi sonucunda duyarlılık analizinin bir versiyonu elde edilebilmektedir. Bazı öğrenme makineleri ve bazı amaç fonksiyonları için her seferinde birini dışarıda bırakma çapraz geçerlilik hatası için analitik formülasyonlar kesin veya yaklaşık olarak bilinmektedir. Bu yöntem Rivals ve Personnaz (2003) tarafından doğrusal en küçük kareler modeli için uygulanmıştır. Rakotomamonjy (2003) tarafından önerilen model ise nonlinear destek vektör makinelerinin (Boser ve diğ., 1992; Vapnik, 1998) standart teknikler ile eğitilmesi ve özellik seçiminin geriye doğru eleme yöntemi ile yinelemeli özellik elemesine (recursive feature elimination) (Guyon ve diğ., 2002) benzer şekilde yapılmasıdır. Ancak değişken derecelendirme kriteri amaç fonksiyonunun duyarlılığı değil her seferinde birini dışarıda bırakma sınırı kullanılarak hesaplanmaktadır.

5.2.3. Doğrudan Amaç Optimizasyonu

Değişken seçimi için amaç fonksiyonunun oluşturulması ve optimize edilmesi alanında önemli aşamalar kaydedilmiştir. Genellikle amaç fonksiyonu birbirini dengeleyen iki terimden oluşmaktadır: uyum iyiliği (maksimize edilmeye çalışılır) ve değişken sayısı (minimize edilmek istenir). Bu yaklaşım, özellikle düzleştirme terimi ile değişken uzayının daraltılması istendiğinde, uyum iyiliği ve düzleştirme terimini içeren iki kısımlı amaç fonksiyonları ile benzerlik göstermektedir. Bu benzerlik Weston ve diğ. (2003) tarafından destek vektör makineleri için doğrusal tahmin ediciler ile sınıflandırma probleminde uygulanmıştır. Düzleştirme terimi $\|\mathbf{w}\|_p^p = \sum_{i=1}^N \mathbf{w}_i^{p/1/p}$ (ℓ_p -norm) şeklinde kullanılmıştır. Limit durumunda, $p \rightarrow 0$, ℓ_p -norm değişken sayısına eşit olmaktadır. Weston ve diğ. (2003) ayrıca doğrusal destek vektör makineleri algoritmalarında basit bir modifikasyon ile ℓ_0 -norm formulasyonunun yaklaşık olarak çözülebileceğini göstermiştir. İzlenen adımlar aşağıda özetlenmiştir:

- i. Standart doğrusal destek vektör makinelerini ℓ_1 -norm veya ℓ_2 -norm düzleştirme terimi kullanarak eğit.
- ii. Girdi değişkenlerini elde edilen ağırlık vektörünün elemanlarının mutlak değerleri ile çarparak yeniden ölçeklendir.
- iii. Yakınsama sağlanıncaya kadar ilk iki adımı tekrarla.

Bu yöntem en küçük $|\mathbf{w}_i|$ değerine dayanan geriye doğru eleme yöntemini anımsatmaktadır. Bu yöntem ile başarılı sonuçlar elde edilebilmesinde değişken normalizasyonu önemli bir etkidir.

Weston ve diğ. (2003) önerilen algoritma ile ℓ_0 -normu yaklaşık olarak minimize edilmesine rağmen uygulamada daha iyi genelleştirme özelliğine sahip olduğunu belirtmiştir. Çünkü aksi taktirde optimizasyon problemi birden fazla çözüme sahip olacağından düzleştirme terimi yeterli etkiyi gösteremeyecektir. İlave düzleştirme terimi ihtiyacına Perkins ve diğ. (2003) de değinmiş, büyük sayıda değişken sayıları için ceza katsayısını da modele ekleyerek üç kısımlı bir amaç fonksiyonunun kullanılmasını önermişlerdir. Amaç fonksiyonunun optimize edilmesi için ise ileri doğru seçme yöntemi kullanılmıştır.

Bi ve diğ. (2003) ℓ_1 -norm destek vektör makinelerinin tekrarlayan çarpımsal güncellemeler olmaksızın kullanılmasını önermiş ve geliştirilen yöntemin uygulama sonuçlarında yeteri kadar ağırlığın sifıra eşitlenmesinde başarılı olduğunu göstermişlerdir. Geriye doğru eleme yöntemi kullanılarak değişken sayısının daha fazla azaltılması da mümkün olabilmektedir.

5.2.4. Alt Küme Seçimi için Filtreleme Yöntemleri

Filtreleme yöntemlerinin uygulamasının daha hızlı olduğu savunulsa da gömülü yöntemler için geliştirilen etkili algoritmalar bu farkı oldukça kapatmıştır. Ayrıca karşılıklı bilgi kriterine dayanan filtreleme yöntemleri değişkenler için genel bir seçim yapmakta, verilen makine öğrenme algoritmasına özgü sonuçlar sunmamaktadır. Bununla birlikte filtreleme yöntemi veri boyutunun azaltılması ve aşırı uyum probleminin önlenmesi maksadıyla ön işleme adımı olarak kullanılabilir.

Bu bağlamda bir sarmalayıcı veya gömülü yöntemi doğrusal tahmin edici ile filtreleme amacı ile kullanmak ve daha sonrasında seçilen değişkenler üzerinde daha karmaşık olan nonlinear tahmin edicinin eğitilmesi önerilmektedir. Bu yaklaşımın bir örneği Bi ve diğ. (2003) tarafından uygulanmıştır, değişken seçimi için doğrusal ℓ_1 -norm destek vektör makineleri, tahmin için ise nonlinear ℓ_1 -norm destek vektör makineleri kullanılmıştır.

5.3. DESTEK VEKTÖR REGRESYON İÇİN DEĞİŞKEN SEÇİMİ

Son zamanlarda makine öğrenme algoritmalarını kullanan regresyon modellerine gösterilen ilgi bu modellerin sahip oldukları özellikler sebebiyle oldukça artmıştır. Makine öğrenme yöntemleri en küçük kareler regresyon modelinin varsayımlarına bağlı olmaksızın, hataların normal dağılması gibi, tahmin performansını geliştirmektedir. Çekirdek-tabanlı makine öğrenme yöntemlerinden destek vektör regresyon da yerel minimuma takılmama ve iyi genelleştirme özelliği gibi pek çok avantaja sahiptir. Ancak bu yöntem değişkenlerin / özelliklerin önem derecelerine ilişkin bilgi vermemektedir.

Destek vektör sınıflandırma için önceki kısımlarda açıklandığı üzere çok sayıda yöntem önerilmiştir. Ancak destek vektör regresyon için kullanılacak çok az sayıda değişken

seçimi yaklaşımı mevcuttur. Bu kısımda destek vektör regresyon için uygulanabilecek değişken seçimi yöntemlerine değinilecektir.

Guajardo ve diğ. (2006) tarafından özellik ve model seçimi için önerilen yöntem şu şekilde özetlenebilir: İlk olarak genel koşullar altında iyi sonuçlar veren hata duyarlılığı, ceza katsayısı ve çekirdek fonksiyonu parametresi değerleri hesaplanır. Daha sonra bu temel parametre değerleri ve ileri doğru seçim ile sarmalama bir özellik seçimi kullanılarak tahmin edici değişkenler kümesi elde edilir. Son olarak belirlenen değişkenler kullanılarak model kurma problemine geri dönülür ve optimum parametre değerlerini bulmak için temel parametre değerleri bölgesinde grid arama tekniği uygulanır. Böylece tahmin edici model, optimum parametre değerleri ve değişkenler kümesi ile belirlenmiş olur.

Maldonado ve Weber (2010) destek vektör makineleri için gömülü bir özellik seçimi yöntemi önermişler ve Gaussian çekirdek fonksiyonu üzerinde bir uygulamasını sunmuşlardır. Bu yaklaşımın temeli ilgili çekirdek fonksiyonunun optimize edilmesi ve regresyon modelinin genelleştirme performansı üzerinde doğrudan etkisi olan değişkenlerin belirlenmesi ile tahmin performansının iyileştirilmesine dayanmaktadır. Çekirdek fonksiyonunun optimizasyonu ve özellik eleme için eğimli düşüş (gradient descent) yaklaşımı kullanılarak destek vektör regresyonun dual formülasyonunda özelliklerin kullanımı cezalandırılmaktadır. Yöntemin amacı uyum iyiliği parametresi, genelleştirme ve özellik seçimi ile çözülmek istenen problem için en uygun ve en küçük boyuta sahip Gaussian çekirdek fonksiyonunun bulunmasıdır.

Yang ve Ong (2010) olasılık tahminlerini kullanarak destek vektör regresyon için sarmalayıcı-tabanlı bir özellik seçimi yöntemi önermişlerdir. Önerilen yöntem destek vektör regresyon tahmininin özellik uzayında araştırılan özellik modelde ve model dışında iken hesaplanan koşullu olasılık fonksiyonlarının farkının toplanması ile özelliğin önemini hesaplamaktadır.

Guajardo ve diğ. (2010) ileri doğru seçim ile bir sarmalayıcı yöntem önermişlerdir. Önerilen özellik seçimi yöntemi her bir bireysel değişkenin tek tahmin edici olarak alınarak model kurulması ile başlar. İkinci adımda en iyi bireysel tahmin ediciler diğer değişkenler ile birleştirilir. En iyi ikili tahmin edici grubu bir sonraki adım için saklanır

ve en iyi maksimum tahmin edici sayısına ulaşıldığında durulur. Sonuç olarak seçilen en iyi değişkenler grubu destek vektör regresyon modelinde kullanılır.

Zhao ve Magoules (2011) tarafından her bir değişkenin derecelendirilmesi için iki özellik seçimi önerilmiş ve değişkenler derecelerine göre filtrelenmiştir. İlk yöntem her bir değişkene bu değişken ve yanıt değişkeni arasındaki korelasyon katsayısına bağlı olarak bir skor vermektedir. İkinci yöntemde ise her bir değişkene bir ağırlık atanmakta ve tüm değişkenlerin ağırlık vektörü eş zamanlı olarak eğimli yükseliş (gradient ascent) tekniği ile optimize edilmektedir.

Karagiannopoulos ve diğ. (2007) tarafından sarmalama özellik seçimi yöntemlerinin regresyon algoritmaları üzerinde uygulama sonuçlarının bir karşılaştırılması sunulmuştur.

6. UYGULAMA

Bu bölümde çekirdek tabanlı makine öğrenme algoritmalarından destek vektör regresyon yönteminin uçak yakıt tüketiminin tahmin edilmesi üzerinde uygulama sonuçları sunulacaktır. İlk kısımda yakıt tasarrufunun ekonomik ve çevresel açıdan önemine değinilecektir. Kurulan modeller gerçek veri seti üzerinde test edilmiştir. Türkiye'deki en büyük havayolu şirketlerinin birisinden elde edilen veri seti üzerinde uygulama sonuçları sunulmuştur. Yakıt tüketiminin modellenmesinde Uçuş Veri İzleme (UVİ (Flight Data Monitoring - FDM)) sistemlerinde kaydedilen ve uçağın durumunu gösteren anlık ölçümler kullanılmıştır. Sonuçların yorumlanabilirliğinin artırılması amacıyla Uçuş Veri İzleme sistemlerindeki kavram ve faaliyetlere ilişkin özet bilgi sunulacaktır. Daha sonra, modelde kullanılan değişkenler hakkında açıklayıcı bilgiler verilecek ve son olarak da model kurulurken göz önünde bulundurulmuş uçuşun safhaları konusu kısaca ele alınacaktır.

İkinci kısımda verinin hazırlanmasında uygulanan işlemler açıklanacaktır. Daha sonraki kısımlarda sırası ile tek çekirdek fonksiyonu ve çoklu çekirdek algoritmaları kullanılarak oluşturulan model sonuçları yer almaktadır. Son kısımda ise destek vektör regresyon yönteminin diğer regresyon modelleri ile karşılaştırılması sunulacaktır.

Destek vektör regresyon için LibSVM (A Library for Support Vector Machines) yazılımının MATLAB arayüzü (interface) kullanılmıştır (Chang ve Lin, 2011).

6.1. HAVACILIKTA YAKIT TÜKETİMİ

6.1.1. Havacılıkta Yakıt Tasarrufu: Çevresel ve Ekonomik Bakış Açısı

Yakıt tasarrufu hem ekonomik hem de çevresel açıdan havayolu şirketleri için büyük önem arz eden bir çalışma sahasıdır.

Çoğu bilim adamı küresel ısınmanın en büyük nedeninin sera gazı etkisi olduğunu değerlendirmektedir. Sera etkisi yaratan gazların başında da karbondioksit gelmektedir. Havacılık sektörü en hızlı büyüyen sera gazı salınımı kaynaklarından birisidir.

Devlet Hava Meydanları İşletmesi (DHMİ) verilerine göre Türkiye'deki havayolu şirketleri tarafından yapılan ticari uçuş sayısı 2013 yılında %11,9 artışla 1.059.391'e yükselmiştir. Bu uçuşların 554.166 adedini iç hat uçuşları, 505.225 adedini ise dış hat uçuşları oluşturmaktadır.

İstanbul-New York arası uçuş yapan bir kişinin neden olduğu emisyon miktarı bir kişinin bir yıl boyunca ısınma amaçlı yakıt kullanımını sonucu neden olduğu emisyon miktarı ile aynıdır. Avrupa Birliği, Emisyon Ticareti Sistemine kendi hava sahasına gelen ve bu sahadan kalkan tüm havayolu şirketlerinin dâhil edilmesi ile ilgili çalışmalarını sürdürmektedir. 2020 yılına kadar bu konuda ciddi somut yaptırımlar getirilmesi planlanmaktadır. Bu durum ülkemizin tüm havayolu şirketlerini yakından ilgilendirmektedir.

Bununla birlikte havayolu şirketlerinin maliyetlerinin %30'unu yakıt masrafları oluşturmaktadır. Yakıt tasarrufu için uygulanabilecek teknolojik gelişmelere ve altyapının iyileştirilmesine dayanan stratejiler (uçak filolarının yenilenmesi, yeşil havaalanı projesi, uçuş rotalarının yeniden düzenlenmesi, alternatif yakıt gibi) üzerinde çalışılmaktadır. Ancak bu çalışmaların çoğu uzun zaman ve büyük miktarda yatırım gerektirmektedir. Yakıt tasarrufu sağlanabilecek alanlardan birisi de operasyonel tedbirlerdir.

Yakıt masrafları havayolu şirketleri için önemli bir operasyonel gider kalemi olduğundan titizlikle takip ve kontrol edilmektedir. Havayolu şirketleri tarafından yakıt tüketiminin takip edilmesinde uygulanan prosedür şu şekildedir: Her bir uçuş için planlanan toplam yakıt tüketim miktarı kullanılan yazılım aracılığı ile hava koşulları, uçuş rotası, yolcu sayısı vb. faktörler dikkate alınarak belirlenmektedir. Uçuş sonrasında gerçekleşen toplam

yakıt tüketim miktarı kaydedilmektedir. Planlanan ve gerçekleşen toplam yakıt tüketim miktarları karşılaştırılmakta ve planlanan miktardan daha fazla yakıt tüketilen uçuşlar incelenmektedir.

Aşırı yakıt tüketiminin nedeni pilot kaynaklı, yani uçağın kullanılma şekli, olabileceği gibi bakım ile ilgili bir konu da olabilir ve çözülmediği takdirde güvenlik ile ilgili problemlere yol açabilir. Örneğin uçağın kontrol yüzeyinin konumu uygun değil ise uçak üzerinde fazladan sürüklenme kuvveti oluşturabilir ve bu da aşırı yakıt tüketimine neden olabilir.

Bu çalışmada uçağın anlık yakıt tüketiminin uçuş süresince veri izleme sistemleri tarafından kaydedilen parametrelerin nonlinear bir fonksiyonu olarak modellenebileceği gösterilmiştir. Elde edilen regresyon modelinden faydalanılarak gerçekleşen yakıt tüketiminin beklenen değerin üzerinde olduğu durumlar tespit edilebilir ve aşırı yakıt tüketimine neden olan faktörler incelenerek gerekli önlemlerin alınması ile yakıt tasarrufu sağlanabilir. Uygulanan yöntem çoğu modern uçakta mevcut olan Uçuş Veri İzleme sistemlerinde kaydedilen veriyi kullanmakta; ekstra veri, yeni bir ekipman, mevcut düzenleme ve standartlarda değişiklik veya ilave bir maliyet gerektirmemektedir.

6.1.2. Uçuş Veri İzleme Sistemleri

Bu kısımda Uçuş Veri İzleme (UVİ) sistemlerinde yer alan kavram ve faaliyetlere ilişkin bilgi verilecek, UVİ sistemlerine ilişkin yasal düzenlemeler ana hatları ile sunulacak ve UVİ sistemlerinin faydaları açıklanacaktır.

UVİ sistemleri, Amerika Birleşik Devletlerinde Uçuş Operasyonel Kalite Güvencesi (Flight Operation Quality Assurance - FOQA) sistemleri olarak adlandırılmaktadır, havayolu şirketleri ve diğer uçak operatörlerinin güvenlik ve operasyonel yönetim sistemlerinin ayrılmaz bir parçasıdır.

UVİ, uçuş operasyonlarının kabul edilen normlardan sapmalarının ve istenmeyen trendlerin tespit edilmesi, muhtemel nedenlerin belirlenmesi ve düzeltilmesi amacıyla uçuş verilerinin toplanması ve analizi sürecidir. Trendlerin takip edilmesi ve rapor edilen olaylara neden olan durumların araştırılması ile uçuş operasyonlarının daha iyi anlaşılması yeteneğini sağlamaktadır. Hatalar daha büyük problemlere yol açmadan tespit

edilebilmekte ve eğitimin artırılması, uçuş operasyonlarına ilişkin prosedürlerin değiştirilmesi gibi önleyici ve düzeltici tedbirler geliştirilebilmektedir.

Uluslararası Sivil Havacılık Organizasyonu (International Civil Aviation Organization - ICAO) standartlarına göre sertifikalı maksimum kalkış ağırlığı 27.000 kg`dan fazla olan uçaklarda güvenlik yönetimi sisteminin bir parçası olarak uçuş veri analiz programlarının kurulması ve işletilmesi gerekmektedir (ICAO Standards and Recommended Practices (SARPs)).

UVİ sürecini oluşturan faaliyetler Şekil 6.1`de sunulmuştur. Veri toplama UVİ sürecinin ilk adımıdır, uçakta bulunan sensörler vasıtası ile gerçekleştirilmekte ve uçak motorunun çalışmaya başladığı andan itibaren motor kapatılana kadar devam etmektedir. Toplanan veriler uçakta bulunan kayıt sistemlerinde kaydedilmekte ve uçak iniş yaptıktan sonra veri analizinin yapılacağı yer birimlerine transfer edilmektedir. Büyük boyutlarda olan uçuş verileri düzenli olarak kaydedilmektedir, ancak verinin kaydedilmesi yeterli değildir. UVİ sistemlerinin operasyonel etkinliğin artırılmasında etkin bir araç olarak kullanılabilmesi için havayolu şirketleri tarafından kaydedilen verinin düzenli olarak analiz edilmesi gerekmektedir. Veri analizinin amacı uçuş süresince gerçekleşen, önceden belirlenmiş olan kriterlerden sapmaların tespit edilmesidir. Veri analizi; istatistiksel analizler, trend analizleri ve risk değerlendirmelerini içeren özet raporların oluşturulması ile tamamlanmaktadır. Düzenli ve periyodik olarak yapılan veri analizleri faydalı geri besleme sunmakta ve uçuş güvenliği, bakım, uçuş operasyonları, eğitim gibi organizasyon içi kritik alanlarda olduğu gibi hava trafik kontrolü, uçak üreticileri, havaalanları gibi fonksiyonlarda da bilgiye dayalı kararlar alınmasını sağlamaktadır. Önleyici ve düzeltici eylemler geliştirilmekte ve bu eylemlerin etkinlikleri takip edilmektedir.



Şekil 6.1. Uçuş Veri İzleme Süreci (Teledyne Controls Flight Data Monitoring Guide)

UVİ sistemlerinin temel amacı uçuş güvenliğinin artırılması olmakla birlikte operasyonel maliyet etkinliği konusunda da önemli katkılar sağlamaktadır. Normalin dışında yüksek oranlarda yakıt tüketimine neden olan spesifik bir uçağın tespit edilmesi ve gerekli düzenlemelerin yapılması ile büyük miktarda yakıt tasarrufu sağlanması mümkündür.

Türkiye'deki havayolu şirketleri UVİ sistemlerinin faydalarından maksimum seviyede istifade edebilmek için etkin olarak çalışmaktadır ve uçuş güvenliğinin artırılması konusunda önemli ilerlemeler kaydedilmiştir (ICAO Working Paper, 2013).

Bu tezde operasyonel maliyet etkinliğinin artırılması ve karbon ayak izinin minimize edilmesinde UVİ sistemlerinden faydalanılması maksadıyla makine öğrenme algoritmaları kullanılarak yakıt tüketiminin modellenmesi amaçlanmaktadır.

Havayolu şirketleri tarafından yakıt tüketiminin takip edilmesinde uygulanan mevcut prosedür şu şekildedir: Her bir uçuş için planlanan toplam yakıt tüketim miktarı kullanılan yazılım aracılığı ile hava koşulları, uçuş rotası, yolcu sayısı vb. faktörler dikkate alınarak belirlenmektedir. Uçuş sonrasında gerçekleşen toplam yakıt tüketim miktarı kaydedilmektedir. Planlanan ve gerçekleşen toplam yakıt tüketim miktarları karşılaştırılmakta ve planlanan miktardan daha fazla yakıt tüketilen uçuşlar incelenmektedir.

Bu çalışmada uçağın anlık yakıt tüketiminin uçuş süresince UVİ sistemleri tarafından kaydedilen parametrelerin nonlinear bir fonksiyonu olarak modellenebileceği gösterilmiştir. Elde edilen regresyon modelinden faydalanılarak gerçekleşen yakıt tüketiminin beklenen değerin üzerinde olduğu durumlar tespit edilebilir ve aşırı yakıt tüketimine neden olan faktörler incelenerek gerekli önlemlerin alınması ile yakıt tasarrufu sağlanabilir. Bir sonraki kısımda UVİ sistemlerinde kaydedilen parametreler hakkında detaylı bilgi sunulacaktır.

6.1.3. Uçuş Veri Setinin Tanıtılması

Uçuş boyunca anlık yakıt tüketiminin modellenmesinde Türkiye'deki bir havayolu şirketinin Uçuş Veri İzleme (UVİ) sisteminde kaydedilmiş olan verilerden faydalanılmıştır. Havayolu şirketi ile yapılan gizlilik anlaşması nedeniyle havayolu şirketinin adı ve uçuş verileri paylaşılmamaktadır. Modelde kullanılan değişkenlerin değerleri destek vektör regresyon yönteminin başarısı için gerekli olması nedeniyle ve veri kaynağını korumak maksadıyla [0, 1] aralığında ölçeklendirilmiştir.

UVİ sistemlerinde uçuş süresince her saniye yüzün üzerinde parametre kaydedilmektedir. Veri setimiz aynı rotada yapılmış 50 iç hatlar ve 50 dış hatlar uçuşundan oluşmaktadır. Uçak tipi aynıdır. Uçaklar kuyruk numaraları ile tanımlanmaktadır. Yakıt tüketimini etkileyen parametreler havayolu şirketinin güvenlik ve UVİ uzmanının desteği ile belirlenmiştir.

Parametreler hakkında açıklayıcı bilgiler aşağıda sunulmuştur:

Girdi Değişkenleri:

Toplam hava sıcaklığı (total air temperature): Toplam hava sıcaklığı uçağın yüzeyi üzerine monte edilmiş sıcaklık probu ile ölçülür. Ölçü birimi C derecedir.

Rüzgar hızı (wind speed): Ölçü birimi knots (deniz mili, yaklaşık 0.5 metre/saniye)'dir.

Rüzgar yönü (wind direction): Ölçü birimi derecedir, -180 ve +180 arasında değer alır.

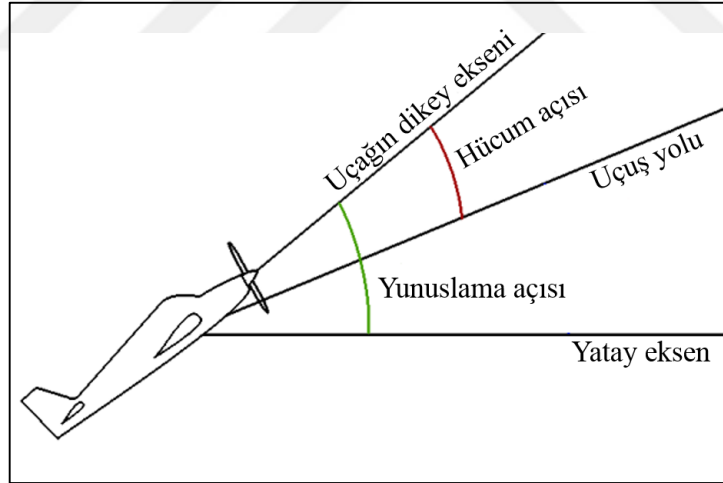
Hesaplanan hava hızı (computed air speed): Uçağın yere göre hızı değil, havaya göre hızını ifade etmektedir. Yani rüzgârın etkisini de içermektedir. Ölçü birimi knots (deniz mili, yaklaşık 0.5 metre/saniye)'dir.

Standart yükseklik (standard altitude): Deniz seviyesinden olan yüksekliktir. Ölçü birimi feet (adım, yaklaşık 30 cm)'dir.

Radyo yükseklik (radio altitude): Uçakta bulunan radyo yükseklikölçer cihazı, uçaktan radyo sinyallerinin gönderilmesi ve zeminden geri yansıtılan sinyallerin alınması arasında geçen süreyi ölçerek uçak ile uçağın altında bulunan arazi bölgesi arasındaki mesafeyi hesaplar. Yaklaşma (approach) ve iniş (landing) için özellikle görüş şartlarının kısıtlı olduğu zamanlarda kullanılır. Ölçü birimi feet (adım, yaklaşık 30 cm)'dir.

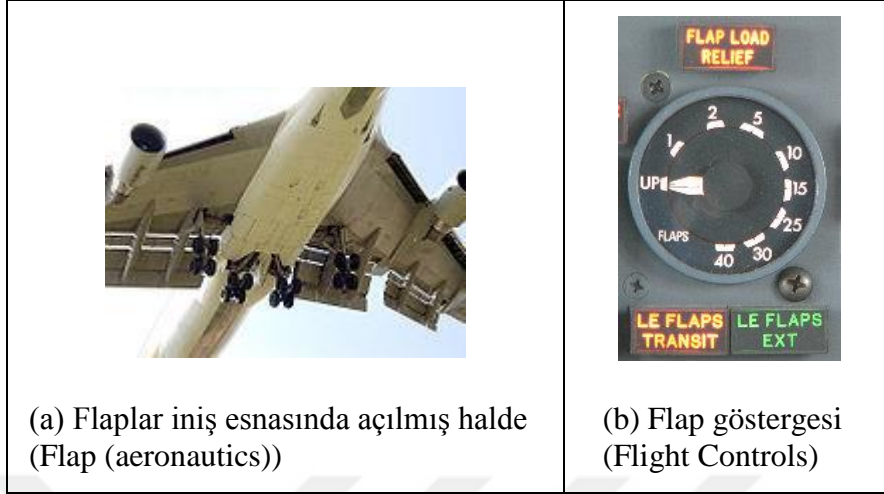
Hücum açısı (sağ ve sol) (angle of attack - right and left): Hücum açısı yaklaşan hava veya rüzgâr ile uçağın kanatlarındaki referans hat arasındaki açıdır. Bu referans hat kanatların ön ve arka kenarlarını ortada bir noktada birleştiren hattır. Ölçü birimi derecedir.

Yunuslama açısı (pitch angle): Uçağın dikey eksenini ve ufuk arasındaki açıdır. Ölçü birimi derecedir. Hücum açısı ve yunuslama açısı Şekil 6.2'de gösterilmiştir.



Şekil 6.2. Hücum açısı ve yunuslama açısı (Boeing - Aero Magazine)

Flap konfigürasyonu (flap configuration): Flaplar uçak kanatlarının arka kenarında bulunurlar. Taşıma kuvvetini ve sürüklemeyi arttırmaları. Flaplar uçuşun özellikle kalkış ve iniş gibi düşük hızlarda daha yüksek taşıma kuvvetine ihtiyaç duyulan safhalarında kullanılırlar, açık duruma getirilirler. Uçaklarda flapların birden fazla ayar düzeyi (açısı) bulunur. Yüksek flap ayarları düşük flap ayarlarına göre daha fazla yakıt tüketir. Şekil 6.3'te flapların açık olduğu durum ve flap göstergesi sunulmuştur.



Şekil 6.3. Flap konfigürasyonu

İniş takımları durumu (landing gear status): İniş takımlarının açık olup olmamasına göre yer (full ground) ve hava (full air) şeklinde iki değer almaktadır.

Engine rpm (revolution per minute): Motorun dakikada yaptığı devir sayısıdır. Her iki motor için % olarak verilmiştir. 0 ile 100 arasında değerler almaktadır.

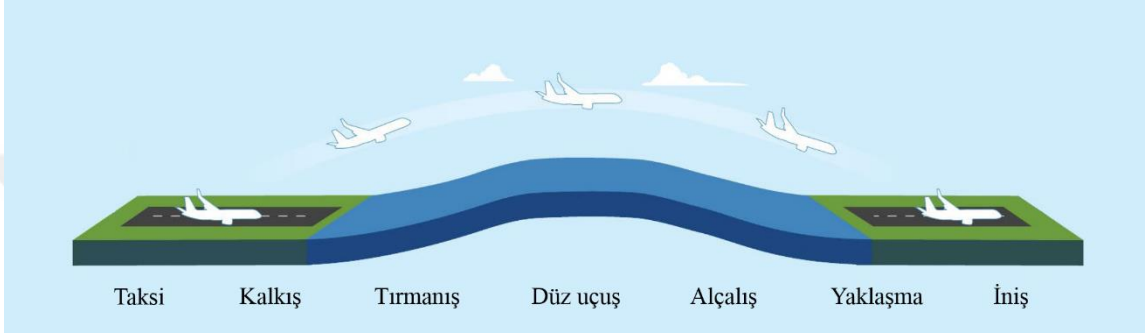
Uçağın toplam ağırlığı (aircraft gross weight): Uçağın o andaki toplam ağırlığıdır. Yakıt tüketiminden dolayı uçağın toplam ağırlığı uçuş süresince azalmaktadır. Ölçümü lbs (pound, yaklaşık 0,5 kg)'dır. Bazı uçuşlarda her saniye için veri mevcut değildir. Bunun nedeni kaydedilme aralığının farklı olmasıdır. Yaklaşık olarak dakikada bir defa kaydedilmiştir.

Çıktı Değişkeni:

Yakıt tüketimi (fuel consumption): Her iki motor için birikimli yakıt tüketim miktarları mevcuttur. İki motor tarafından tüketilen yakıt miktarları toplanmıştır. Her saniye için kaydedilen değer bir öncekinden çıkarılarak o saniyede tüketilen yakıt miktarı bulunmuştur. İki motor tarafından saniyede tüketilen toplam yakıt miktarı modelin çıktı değişkeni olarak alınmıştır.

6.1.4. Uçuşun Safhaları

Uçağın yakıt tüketimi uçuşun safhaları ile yakından ilişkili olduğundan bu kısımda uçuşun safhaları ile ilgili özet bilgi verilecektir. Uçuşun safhaları taksi (taxi), kalkış (take-off), tırmanış (climb), düz uçuş (cruise), alçalış (descent), yaklaşma (approach) ve iniş (landing) olarak tanımlanmaktadır. Uçuşun safhaları Şekil 6.4'te gösterilmiştir.



Şekil 6.4. Uçuşun safhaları (NextGen Phases of Flight, 2013)

Taksi, uçağın yerdeki hareketini ifade etmektedir. Kalkış, taksi hareketinden sonra pistin başlangıç noktasında duran uçağın uçuşa geçiş safhasıdır. Uçak motorları kalkışta tam güçte çalıştırılır. Kalkıştan sonra uçakların güvenli ve ekonomik uçuş için belirli bir yüksekliğe kadar (genellikle 30.000 feet (adım) ya da 10 km) tırmanması ve düz uçuş safhasına geçmesi gerekmektedir. Tırmanış safhasının başlangıcında iniş takımları kapatılmaktadır. Düz uçuş, uçuşun en yakıt ekonomik safhasıdır. Düz uçuşta geçirilen süre ve yakıt tüketim miktarı uçuş mesafesine bağlı olup, iç ve dış hatlar uçuşlarında farklılık göstermektedir. Uzun mesafe ticari yolcu uçaklarında düz uçuş hızı genellikle 475-500 knots (878-926 km/sa)'dir. Alçalış safhasında uçağın yüksekliği azalmaktadır ve bu safhada motorlar düşük güç ayarlarında işletilmektedir. İniş safhasının başlangıcında, uçağın yavaşlamasına yardımcı olması amacıyla, iniş takımları açılmaktadır.

6.2. VERİNİN HAZIRLANMASI

Veri setinde 50 iç hatlar uçuşu (İstanbul-İzmir) ve 50 dış hatlar uçuşu (İstanbul-Amsterdam) yer almaktadır. Uçuş süreleri hava trafik kontrolü, rötar vb. nedenlerden dolayı farklılık göstermektedir. Uçuş süre ve mesafelerine ilişkin özet bilgi Çizelge 6.1'de verilmektedir.

Çizelge 6.1. Uçuş süre ve mesafeleri

	Uçuş süresi			Uçuş mesafesi
	Ortalama	Minimum	Maksimum	
İç hatlar	3.853 sn. (64 dk.)	2.196 sn. (37 dk.)	5.648 sn. (94 dk.)	330 km.
Dış hatlar	13.305 sn. (3 sa. 42 dk.)	10.784 sn. (3 sa.)	15.688 sn. (4 sa. 21 dk.)	2.208 km.

Uluslararası Sivil Havacılık Organizasyonu tarafından geliştirilen metodoloji ile uçak yolculuğundan kaynaklanan karbondioksit salınımı miktarlarına da ulaşılabilmektedir (ICAO Carbon Emissions Calculator Methodology, 2014). Örneğin bu çalışmada analiz edilen rotadaki bir iç hatlar uçuşu için yolcu başına ortalama 63,26 kg. karbondioksit, bir dış hatlar uçuşunda ise yolcu başına ortalama 184,56 kg. karbondioksit oluşmaktadır.

İlk 25 iç hatlar uçuşu eğitim seti olarak kullanılmıştır. Eğitim verisinin boyutu 97.536 x 12'dir. Elde edilen model ilk olarak kalan 25 iç hatlar uçuşu (95.104 gözlem) üzerinde test edilmiştir. Daha sonra modelin rotaya spesifik olup olmadığının değerlendirilmesi amacıyla 50 dış hatlar uçuşu üzerinde test yapılmıştır.

Modelde 12 açıklayıcı değişken ve 1 yanıt değişkeni bulunmaktadır. Açıklayıcı değişkenlerin listesi Çizelge 6.2'de sunulmuştur. Yanıt değişkeni iki motorun saniyedeki toplam yakıt tüketim miktarıdır.

Çizelge 6.2. Modelde kullanılan açıklayıcı değişkenler

Nu.	Değişken adı	Nu.	Değişken adı
1	Toplam hava sıcaklığı	7	Hücum açısı (sol)
2	Rüzgâr hızı	8	Yunuslama açısı
3	Rüzgâr yönü	9	Flap konfigürasyonu
4	Hesaplanan hava hızı	10	İniş takımları durumu
5	Standart yükseklik	11	Motorun dakikadaki devir sayısı-1
6	Hücum açısı (sağ)	12	Motorun dakikadaki devir sayısı-2

Veri setindeki tek kategorik deęişken iniş takımlarının durumudur (landing gear status). Bu deęişken iniş takımları açık olduęunda (full ground) 0; kapalı olduęunda (full air) 1 olacak şekilde sayısal deęişkene dönüştürülmüştür.

Destek vektör regresyon yönteminin başarısını önemli ölçüde etkileyen bir konu parametre deęerlerinin uygun seçimidir. Örnek olarak sıklıkla kullanılan Gaussian (radyal tabanlı fonksiyon (radial basis function-RBF)) çekirdek fonksiyonu gösterilebilir.

$$k(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2} \quad (6.1)$$

Gaussian çekirdek fonksiyonunda γ parametresi araştırmacı tarafından belirlenmektedir. Küçük γ deęerleri aşırı uyuma neden olabilecektir. Bununla birlikte Eşitlik (3.22)'de verilen, f fonksiyonunun düzlüğü ve tolere edilecek hata miktarı arasında dengeyi sağlayan C ceza katsayısı da yine araştırmacı tarafından belirlenmektedir. Gaussian çekirdek fonksiyonunda C ve γ parametrelerinin seçimi için yaygın olarak kullanılan yöntem çapraz geçerlilik (cross-validation) yöntemidir. Ancak destek vektör regresyonda optimize edilmesi gereken parametre sayısı fazla olduęundan çapraz-geçerlilik yöntemi etkinliğini yitirmektedir. Bu çalışmada parametre seçimi için çoklu çekirdek algoritmalarından faydalanılacaktır.

Regresyon tahmin sonuçlarının deęerlendirilmesinde performans ölçüsü olarak Hata Kareler Ortalaması (HKO , Mean Squared Error - MSE) deęeri kullanılmıştır.

$$HKO = \frac{1}{N} \sum_{i=1}^N (f(\mathbf{x}_i) - y_i)^2 \quad (6.2)$$

Bununla birlikte aynı alanda yapılmış çalışmaların sonuçları ile karşılaştırma yapılabilmesi maksadıyla Normalleştirilmiş Hata Kareler Ortalamasının Karekökü ($NHKOK$, Normalized Root Mean Squared Error - $NRMSE$) deęeri de hesaplanmıştır. Kullanılan formülasyon aşağıda verilmiştir:

$$NHKOK = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N (f(\mathbf{x}_i) - y_i)^2}}{\text{var}(y)} \quad (6.3)$$

6.2.1. Veri Ölçeklendirme

Veri ölçeklendirme makine öğrenme algoritmaları için çok önemli bir adımdır ve hem hesaplama süresi hem de model başarısını etkilemektedir. Özellikle de bazı değişkenlerin diğerlerine göre çok büyük veya küçük değerlere sahip olması durumunda veri ölçeklendirme tahmin edicinin performansını önemli ölçüde arttırmaktadır. Bu kısımda yaygın olarak kullanılan veri ölçeklendirme yöntemlerine ilişkin özet bilgi sunulacaktır.

Standartlaştırma (z -skor normalleştirme) ile değişkenler 0 ortalama ve 1 standart sapmaya sahip olacak şekilde yeniden ölçeklendirilirler. Standart skorlar (z-skorları) aşağıdaki şekilde hesaplanırlar:

$$z = \frac{x - \mu}{\sigma} \quad (6.4)$$

Standartlaştırma farklı birimlere sahip ölçümlerin karşılaştırılmasına olanak sağlamakla birlikte makine öğrenme algoritmalarının uygulanabilmesi için gerekli koşullardan birisidir. Örnek olarak destek vektör makinelerinde optimizasyon için kullanılan eğimli düşüş (gradient descent) yöntemi verilebilir. Farklı ölçeklerde olan değişken değerleri söz konusu olduğunda değişken değerleri ağırlıkların güncellenmesinde etkili olduğundan bazı ağırlıklar daha hızlı güncellenebilir.

z-skor normalleştirmeye alternatif bir yöntem ise min-maks normalleştirmedir. Bu yaklaşımda veri sabit bir aralığa (genellikle 0 ve 1) ölçeklendirilmektedir.

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (6.5)$$

Bir diğer yöntem ise kareler toplamı 1 olacak şekilde ile normalleştirmedir. Bu yöntemde normalleştirilmiş vektör aşağıdaki şekilde hesaplanmaktadır:

$$x_{norm} = \frac{x}{\|x\|} \quad (6.6)$$

Burada $\|x\|$ vektörün normudur ve aşağıdaki şekilde hesaplanır:

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2} \quad (6.7)$$

Bu çalışmada veri setindeki değişkenler üzerinde büyük sayısal aralıklara sahip olan özelliklerin küçük sayısal aralıklara sahip olanlar üzerinde baskın olmasını önlemek amacıyla Hsu ve diğ. (2010) tarafından önerilen ve LibSVM’de uygulanabilen ölçeklendirme yöntemi kullanılmıştır. Ayrıca çekirdek matrisinin değerleri özelliklerin iç çarpımlarına bağlı olduğundan ölçeklendirme hesaplamalarda da kolaylık sağlayacaktır. Her bir değişken 0 ve 1 arasında normleştirilmiştir. Eğitim ve test verisi için ölçeklendirme yapılırken aynı yöntem kullanılmıştır.

Bu durum Hsu ve diğ. (2010) tarafından şu şekilde örneklendirilmiştir: Eğitim verisindeki ilk özelliğin minimum ve maksimum değerleri -10 ve +10 olsun ve bu özellik -1 ile +1 arasında ölçeklenmiş olsun. Test verisindeki ilk özelliğin minimum ve maksimum değerleri -11 ve +8 ise, bu özelliğin -1.1 ve +0.8 arasında ölçeklenmesi gerekmektedir.

6.3. TEK ÇEKİRDEK FONKSİYONU KULLANILARAK ELDE EDİLEN SONUÇLAR

Bu kısımda tek çekirdek fonksiyonu kullanılarak Destek Vektör Regresyon (DVR) yöntemi ile elde edilen sonuçlar sunulmaktadır. Uçağın anlık yakıt tüketiminin Uçuş Veri İzleme (UVİ) sistemlerinde kaydedilen parametrelerin nonlinear bir fonksiyonu olarak modellenebileceği gösterilmiştir.

Takip edilen metodoloji şu şekildedir: İlk 25 iç hatlar uçuşu eğitim seti olarak kullanılmıştır. Elde edilen model önce kalan 25 iç hatlar uçuşu üzerinde test edilmiştir. Performans ölçüsü olarak Hata Kareler Ortalaması (*HKO*) kullanılmıştır. Bununla birlikte bu alanda yapılmış diğer çalışmalarda elde edilen sonuçlar ile karşılaştırma yapılabilmesi amacıyla Normleştirilmiş Hata Kareler Ortalamasının Karekökü (*NHKOK*) değerleri de hesaplanmıştır. Aşırı uyum probleminin kontrolü amacıyla, aynı model, modelin kurulmasında kullanılan eğitim seti üzerinde de test edilmiştir. Kurulan regresyon modelinin eğitim ve test verisi üzerinde uygulanması sonucunda elde edilen *NHKOK* değerlerinin oranı hesaplanmıştır. Daha sonra kurulan modelin rotaya spesifik olup olmadığının test edilmesi amacıyla model 50 dış hatlar uçuşu üzerinde ayrı ayrı test edilmiştir. Elde edilen Ortalama *HKO* ve *NHKOK* değerleri sunulmuştur.

DVR yönteminin uygulanmasında en büyük problem sahası çekirdek fonksiyonun seçimi ve optimum parametre değerlerinin belirlenmesidir. Yakıt tüketimi ve UVİ verileri arasındaki ilişki nonlineer olduğundan Gaussian ve Polinomial çekirdek fonksiyonları kullanılarak elde edilen sonuçlar Doğrusal çekirdek fonksiyonuna göre oldukça üstündür. Bu kısımda doğruluk derecesi ve model kompleksliği bakımından yakın performans gösteren Gaussian ve Polinomial çekirdek fonksiyonları ile elde edilen sonuçlara yer verilecektir.

Hata tüpünün genişliği (ϵ), eğitim zamanını ve elde edilen modelin doğruluk derecesini optimize edecek şekilde seçilmiştir. Hata tüpünün genişliği, $\epsilon = 0,1$ için Gaussian ve Polinomial çekirdek fonksiyonları ile elde edilen sonuçlar Çizelge 6.3'te sunulmuştur.

Çizelge 6.3. Hata tüpünün genişliği = 0,1 ile elde edilen sonuçlar ($C = 1, \gamma = 0,1, d = 3$)

Çekirdek fonksiyonu		Gaussian	Polinomial
DV sayısı (DV oranı)		252 (252/97536=0,003)	342 (342/97536=0,004)
25 iç hatlar uçuşu üzerinde test	<i>HKO</i>	0,001226	0,001409
	<i>NHKOK</i>	0,1524	0,1634
Eğitim verisi üzerinde test	<i>HKO</i>	0,001179	0,001239
	<i>NHKOK</i>	0,1520	0,1558
	<i>NHKOK</i> oranı (eğitim / test)	0,99	0,95
50 dış hatlar uçuşu üzerinde test	Ort. (<i>HKO</i>)	0,0012	0,0023
	Ort. (<i>NHKOK</i>)	0,2235	0,3104

Çizelge 6.3.'te en dikkat çekici sonuç destek vektör sayısının çok az olmasıdır. Gaussian çekirdek fonksiyonu eğitim örneklerinin 0,003'ünü, polinomial çekirdek fonksiyonu ise 0,004'ünü destek vektör olarak kullanmıştır. *NHKOK* değerleri iç hatlar uçuşları için %15 civarında, dış hatlar uçuşları için ise Gaussian ve Polinomial çekirdek fonksiyonları ile sırasıyla yaklaşık olarak %20 ve %30 olarak hesaplanmıştır.

Daha yüksek doğruluk dereceli bir model oluşturulmak istendiğinden hata tütünün genişliği 0,01 olacak şekilde küçültülmüştür. Elde edilen sonuçlar Çizelge 6.4'te verilmiştir.

Çizelge 6.4. Hata tütünün genişliği = 0,01 ile elde edilen sonuçlar ($C = 1, \gamma = 0,1, d = 3$)

Çekirdek fonksiyonu		Gaussian	Polinomial
DV sayısı (DV oranı)		18546 (18646/97536=%19)	19253 (19253/97536=%20)
25 iç hatlar uçuşu üzerinde test	<i>HKO</i>	0,000191	0,000223
	<i>NHKOK</i>	0,0602	0,0650
Eğitim verisi üzerinde test	<i>HKO</i>	0,000164	0,000197
	<i>NHKOK</i>	0,0566	0,0621
	<i>NHKOK</i> oranı (eğitim / test)	0,94	0,96
50 dış hatlar uçuşu üzerinde test	Ort. (<i>HKO</i>)	0,000406	0,000357
	Ort. (<i>NHKOK</i>)	0,1324	0,1237

Çizelge 6.4'ten görüleceği üzere, hata tütünün genişliği küçültüldüğünde destek vektör olarak seçilen örnek oranı Gaussian çekirdek fonksiyonu için %19'a, Polinomial çekirdek fonksiyonu için %20'ye yükselmiştir. Bununla birlikte modelin doğruluk derecesi artmış, *NHKOK* değerleri iç hatlar uçuşları için %6 civarında, dış hatlar uçuşları için ise Gaussian ve Polinomial çekirdek fonksiyonları ile sırasıyla yaklaşık olarak %13 ve %12 olarak hesaplanmıştır.

Hata katsayısı daha da küçültülerek 0,001'e eşitlendiğinde model tarafından örneklerin neredeyse tamamı destek vektör olarak kabul edilmekte ve yöntem performans etkinliğini kaybetmektedir. Gaussian çekirdek fonksiyonu için DV sayısı 82377 (82377/97536=%84), Polinomial çekirdek fonksiyonu için DV sayısı 82276 (82276/97536=%84) olmaktadır. DVR yönteminde eğitim süresi ve dolayısıyla modelin

performansı destek vektör sayısına oldukça bağlıdır. Eğitim zamanını ve elde edilen modelin doğruluk derecesini optimize eden parametre değeri $\varepsilon = 0,01$ olarak bulunmuştur.

Regresyon modelinin doğruluk oranının arttırılmasında bir diğer strateji de hata tütünün dışında kalan örnekler üzerinde uygulanacak olan C ceza katsayısının arttırılmasıdır. Burada dikkat edilmesi gereken nokta büyük C değerlerinin aşırı uyum problemine neden olabileceğidir. Bu konunun da göz önünde bulundurulması ve incelenmesi gerekmektedir. Gaussian çekirdek fonksiyonu için farklı C parametre değerleri ile elde edilen sonuçlar Çizelge 6.5'te verilmiştir.

Çizelge 6.5. Gaussian çekirdek fonksiyonu için farklı ceza katsayıları ile elde edilen sonuçlar ($\varepsilon = 0,01$, $\gamma = 0,1$)

Ceza katsayısı		$C = 1$	$C = 10$	$C = 100$
DV sayısı		18646	14168	12944
25 iç hatlar uçuşu üzerinde test	<i>HKO</i>	0,000191	0,000149	0,000143
	<i>NHKOK</i>	0,0602	0,0532	0,0521
Eğitim verisi üzerinde test	<i>HKO</i>	0,000164	0,000130	0,000119
	<i>NHKOK</i>	0,0566	0,0505	0,0484
	<i>NHKOK</i> oranı (eğitim / test)	0,94	0,95	0,93
50 dış hatlar uçuşu üzerinde test	Ort. (<i>HKO</i>)	0,000406	0,000251	0,000333
	Ort. (<i>NHKOK</i>)	0,1324	0,1017	0,1179

Çizelge 6.5'te görüleceği üzere; ceza katsayısı değeri 100'e eşitlendiğinde iç hatlar uçuşları için model hatası düşmekte, ancak dış hatlar uçuşları için hata değeri yükselmektedir. Eğitim ve test verisi üzerinde elde edilen *NHKOK* değerlerinin oranı da düşmekte ve aşırı uyum problemine işaret etmektedir. Bu nedenle Gaussian çekirdek fonksiyonu için ceza katsayısının optimum değeri 10 olarak belirlenmiştir.

Polinomial çekirdek fonksiyonu için farklı ceza katsayısı değerleri kullanılarak elde edilen sonuçlar Çizelge 6.6'da verilmiştir.

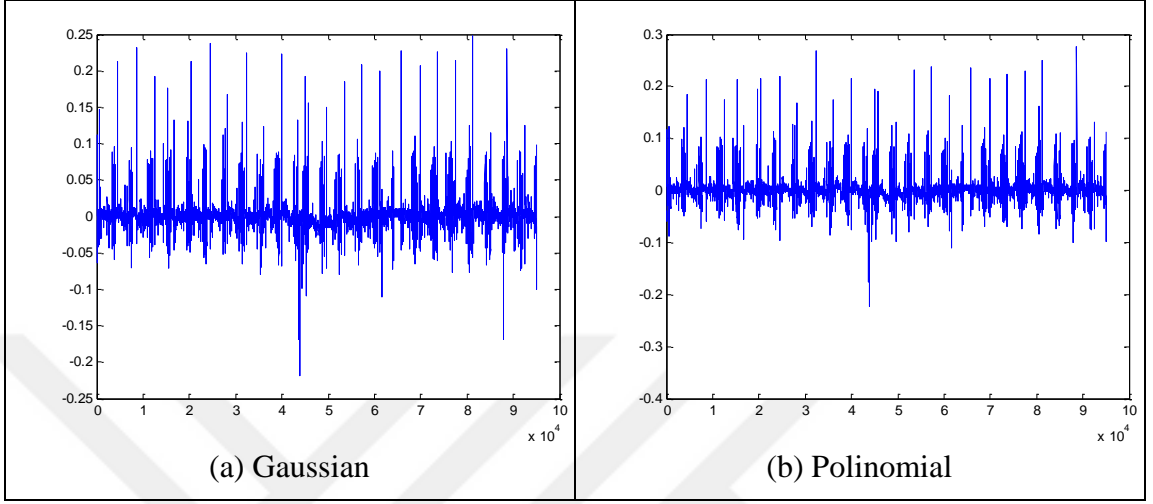
Çizelge 6.6. Polinomial çekirdek fonksiyonu için farklı ceza katsayıları ile elde edilen sonuçlar ($\varepsilon = 0,01$, $d = 3$)

Ceza katsayısı		$C = 1$	$C = 10$	$C = 100$
DV sayısı		19253	16269	15142
25 iç hatlar uçuşu üzerinde test	<i>HKO</i>	0,000223	0,000173	0,000162
	<i>NHKOK</i>	0,0650	0,0573	0,0554
Eğitim verisi üzerinde test	<i>HKO</i>	0,000197	0,000155	0,000134
	<i>NHKOK</i>	0,0621	0,0551	0,0512
	<i>NHKOK</i> oranı (eğitim / test)	0,96	0,96	0,92
50 dış hatlar uçuşu üzerinde test	Ort. (<i>HKO</i>)	0,000357	0,000306	0,000322
	Ort. (<i>NHKOK</i>)	0,1237	0,1130	0,1159

Aynı durum Polinomial çekirdek fonksiyonu için de geçerlidir. Dış hatlar uçuşları üzerinde en yüksek doğruluk derecesine sahip model ceza katsayısının 10 değeri ile elde edilmiştir.

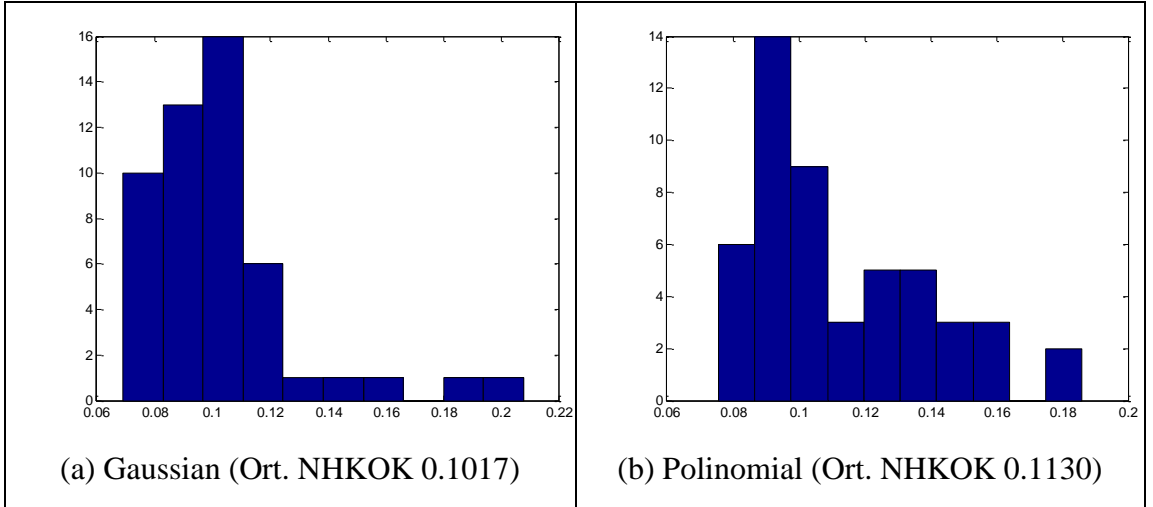
Sonuç olarak $\varepsilon = 0,01$ ve $C = 10$ değerleri ile hem Gaussian hem de Polinomial çekirdek fonksiyonu kullanılarak yüksek doğruluk dereceli modeller oluşturulmuştur. UVİ parametreleri kullanılarak oluşturulan yakıt tüketimi modelinin *NHKOK* değeri iç hat uçuşları için %5, dış hatlar uçuşları için ise %10 civarındadır.

Şekil 6.5'te test verisi olarak kullanılan 25 iç hatlar uçuşu (95.104 gözlem) için artıkların grafiği verilmektedir.



Şekil 6.5. İç hatlar uçuşları için artıklar (*gerçek değer – tahmini değer*)

Şekil 6.6'da 50 dış hatlar uçuşu için Gaussian ve Polinomial çekirdek fonksiyonları ile elde edilen *NHKOK* değerlerinin histogram grafikleri gösterilmektedir.

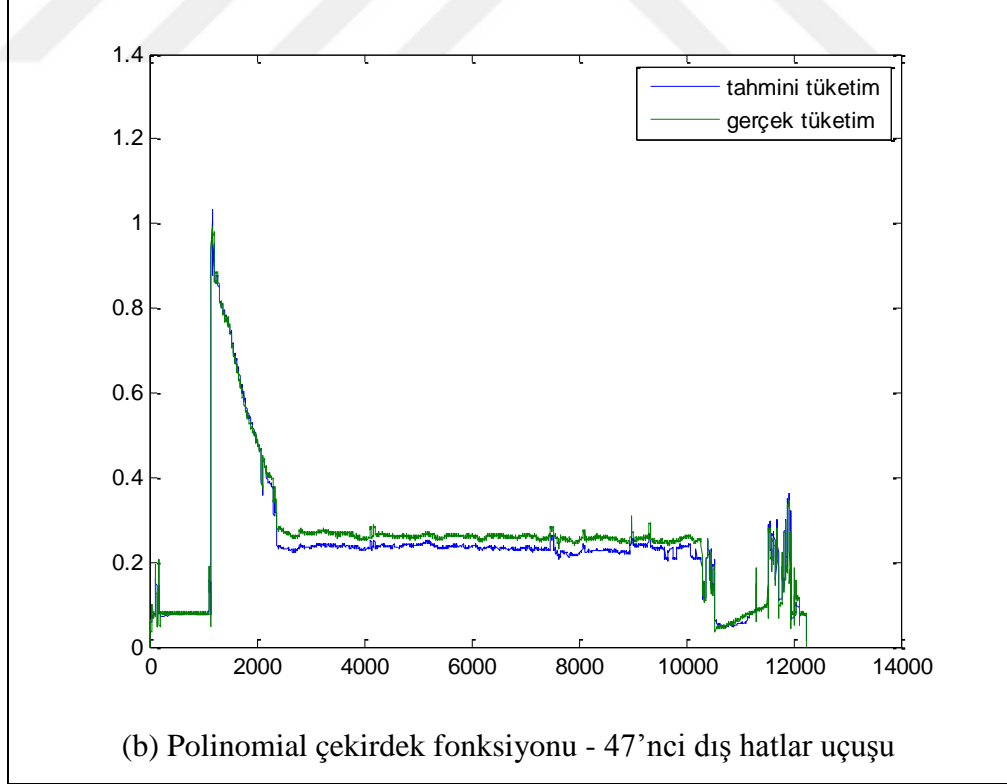
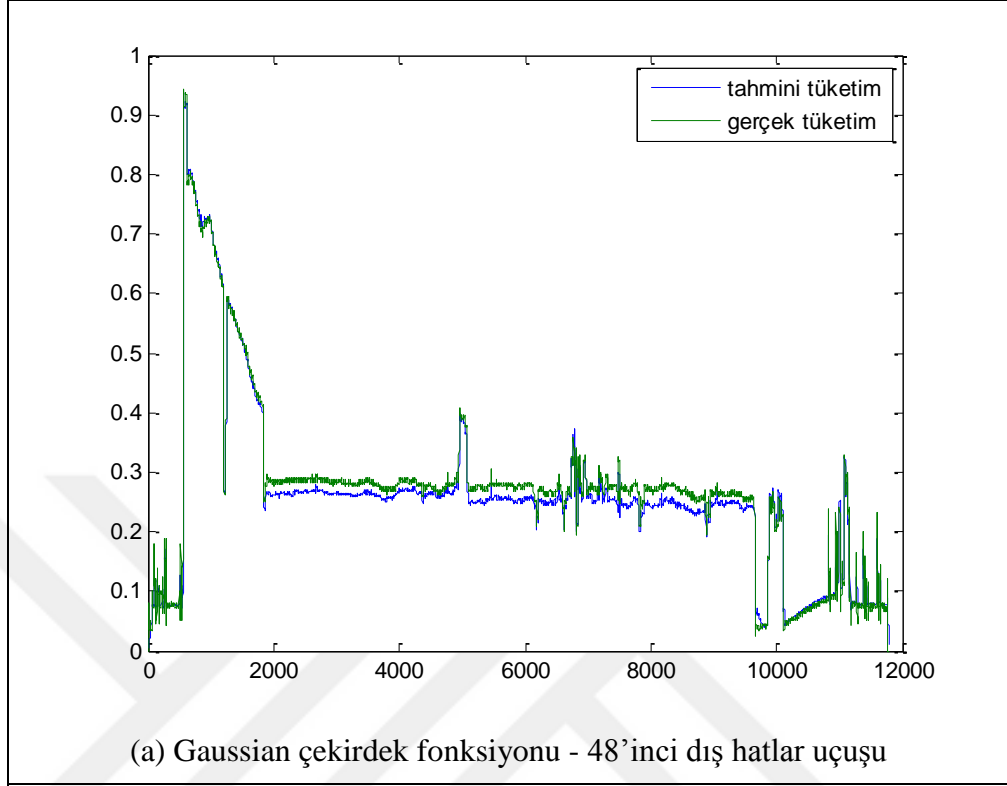


Şekil 6.6. Dış hatlar uçuşları için *NHKOK* değerleri

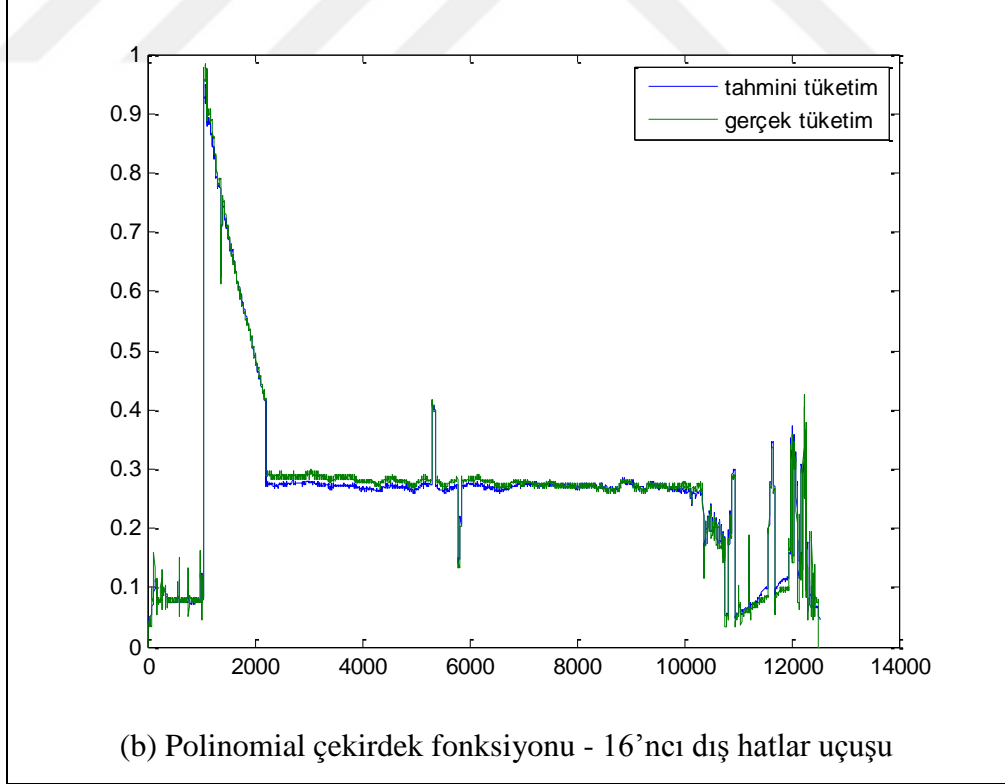
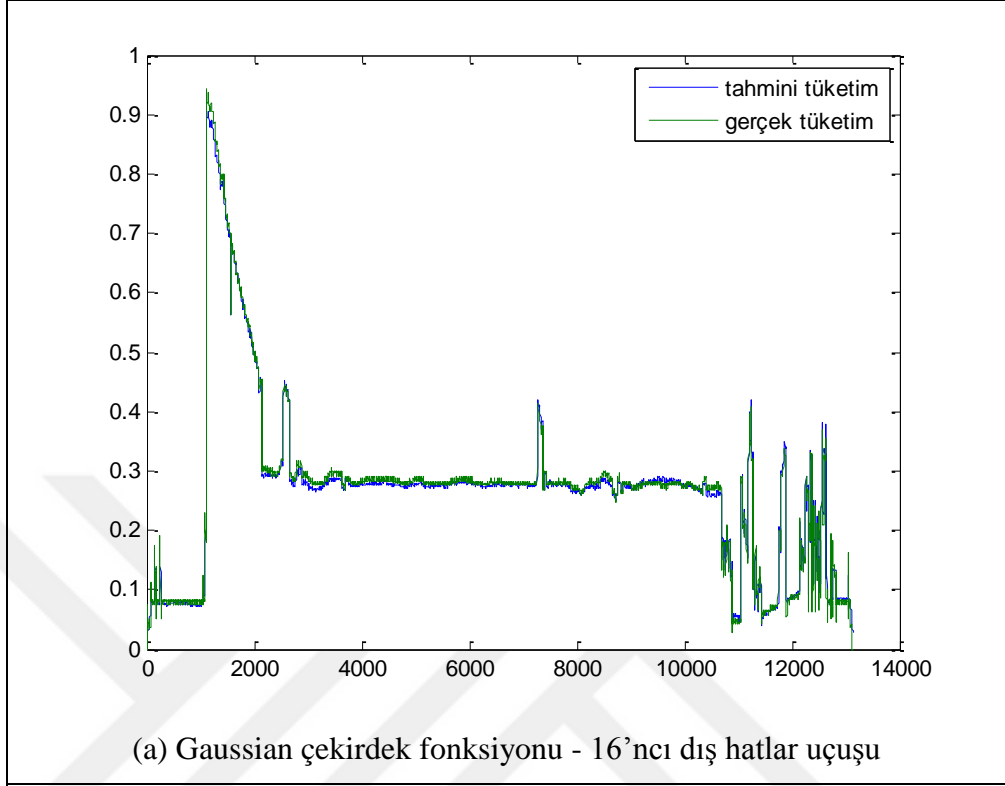
6.3.1. Aşırı Yakıt Tüketimine Sebep Olan Faktörlerin Araştırılması

Aşırı yakıt tüketiminin tespit edilmesinde beklenen değerin ve 0.01 genişliğindeki hata tüpünün üzerinde yakıt tüketimi gerçekleşen uçuş süreleri temel alınmıştır. 50 dış hatlar uçuşunun her biri için uçuş boyunca eşik değerden fazla yakıt tüketilen noktalar hesaplanmıştır. 47 ve 48'inci dış hatlar uçuşları hem Gaussian hem de Polinomial çekirdek fonksiyonu ile kurulan modeller tarafından en fazla aşırı yakıt tüketimi yapılan iki uçuş olarak belirlenmiştir. Bu uçuşlara ait yakıt tüketimi grafikleri Şekil 6.7'de sunulmuştur. Tahmin edilen yakıt tüketimi değerleri mavi renk ile, gerçekleşen yakıt tüketimi değerleri ise yeşil renk ile gösterilmektedir.

Hem Gaussian hem de Polinomial çekirdek fonksiyonu ile kurulan modeller tarafından beklenene en yakın yakıt tüketimi yapılan uçuş 16'ncı dış hatlar uçuşu olarak belirlenmiştir. Bu uçuş için yakıt tüketimi grafiği Şekil 6.8'de sunulmuştur.



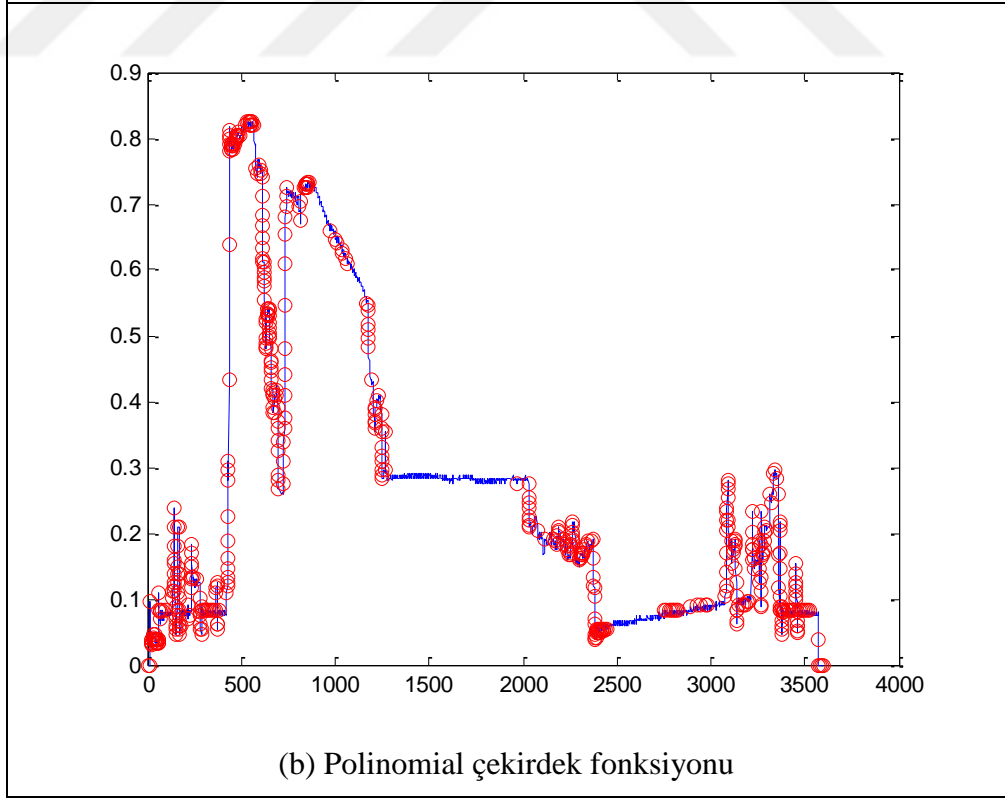
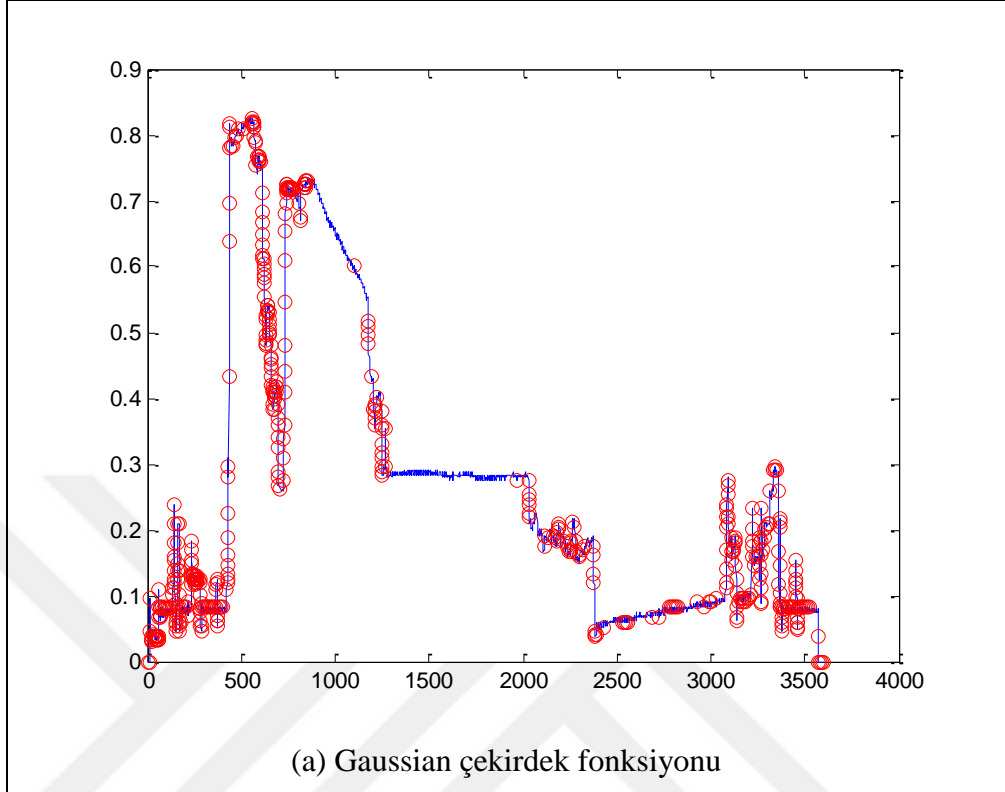
Şekil 6.7. Aşırı yakıt tüketimi yapılan uçuşlara ait yakıt tüketimi grafiği



Şekil 6.8. Normal yakıt tüketimi yapılan uçuşa ait yakıt tüketimi grafiği

Anomali yani aşırı yakıt tüketimi gerçekleşen uçuşların grafikleri incelendiğinde aşırı yakıt tüketiminin çoğunlukla düz uçuş safhasında meydana geldiği görülmektedir. Düz uçuş safhasında yakıt tüketimi üzerinde en etkili olan iki parametre yükseklik ve hızdır. UVİ sistemlerinde kaydedilen parametre değerleri incelendiğinde aşırı yakıt tüketildiği tespit edilen 47 ve 48'inci uçuşlarda düz uçuş safhasındaki uçuş yüksekliğinin 16'ncı uçuştan daha fazla olduğu görülmüştür. Bununla birlikte rüzgârın hızı ve yönünün yakıt tüketimi üzerinde önemli derecede etkili olduğu tespit edilmiştir. Şöyle ki 16'ncı uçuşun düz uçuş safhasındaki hızı 47 ve 48'inci uçuşlara göre daha fazladır. Bu durumda; 16'ncı uçuşta daha fazla yakıt tüketiminin gerçekleşmesi beklenirdi. Ancak hesaplanan hava hızı, yere göre hızı göstermemekte ve rüzgârın etkisini de içermektedir. Rüzgâr ile ilgili parametreler incelendiğinde 16'ncı uçuşta rüzgârın hızının 47 ve 48'inci uçuşlara göre oldukça yüksek olduğu ve 47 ve 48'inci uçuşlarda rüzgârın yönünün yakıt tüketimini olumsuz etkilediği görülmüştür.

Eğitim verisindeki ilk uçuşun yakıt tüketim grafiği ile Gaussian ve Polinomial çekirdek fonksiyonları ile elde edilen modellerde kullanılan destek vektörler örnek teşkil etmesi açısından Şekil 6.9'da sunulmuştur. x-ekseni uçuş süresini y-ekseni ise yakıt tüketim miktarını göstermektedir. Destek vektörler kırmızı çemberler ile işaretlenmiştir.



Şekil 6.9. Destek vektörler

6.4. ÇOKLU ÇEKİRDEK FONKSİYONU KULLANILARAK ELDE EDİLEN SONUÇLAR

Bir önceki kısımda farklı hata tüpü genişliği ve ceza katsayıları kullanılarak oluşturulan modellerin tahmin performansları karşılaştırılmıştır. Bu kısımda ise model seçimi ve çekirdek fonksiyonu için optimum parametre değerlerinin belirlenmesi (Gaussian çekirdek fonksiyonunda genişlik ve Polinomial çekirdek fonksiyonunun derecesi) konusu incelenecektir.

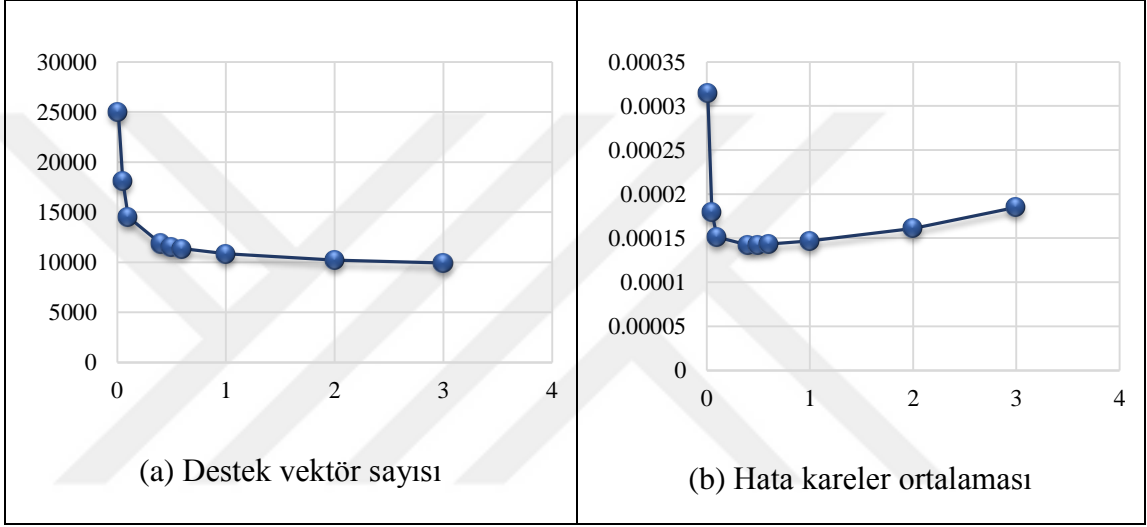
6.4.1. Gaussian Çekirdek Fonksiyonlarının Konik Kombinasyonu

Bu kısımda farklı genişliğe sahip Gaussian çekirdek fonksiyonlarının konik kombinasyonu oluşturulacaktır.

Çizelge 6.7. Gaussian çekirdek fonksiyonu için farklı genişlik değerleri ile elde edilen sonuçlar ($\varepsilon = 0,01$ ve $C = 10$)

Parametre değeri	Destek Vektör sayısı	HKO
0,01	25043	0,000314715
0,05	18146	0,000179227
0,1	14535	0,000151266
0,4	11888	0,000142172
0,5	11555	0,000141825
0,6	11367	0,000143397
1	10855	0,000146789
2	10223	0,000161093
3	9926	0,000184982

Çizelge 6.7’de görüldüğü üzere Gaussian çekirdek fonksiyonunun genişlik parametresinin değeri büyüdükçe destek vektör olarak seçilen örneklerin sayısı dolayısıyla model kompleksliği azalmaktadır. Ancak en düşük hata kareler ortalaması ve en büyük açıklama katsayısı değerini veren parametre değeri 0.5 olarak bulunmuştur. 0.5’ten büyük parametre değerleri için hata kareler ortalamasında artış görülmektedir. Elde edilen bilgiler Şekil 6.10 ile özetlenmiştir.



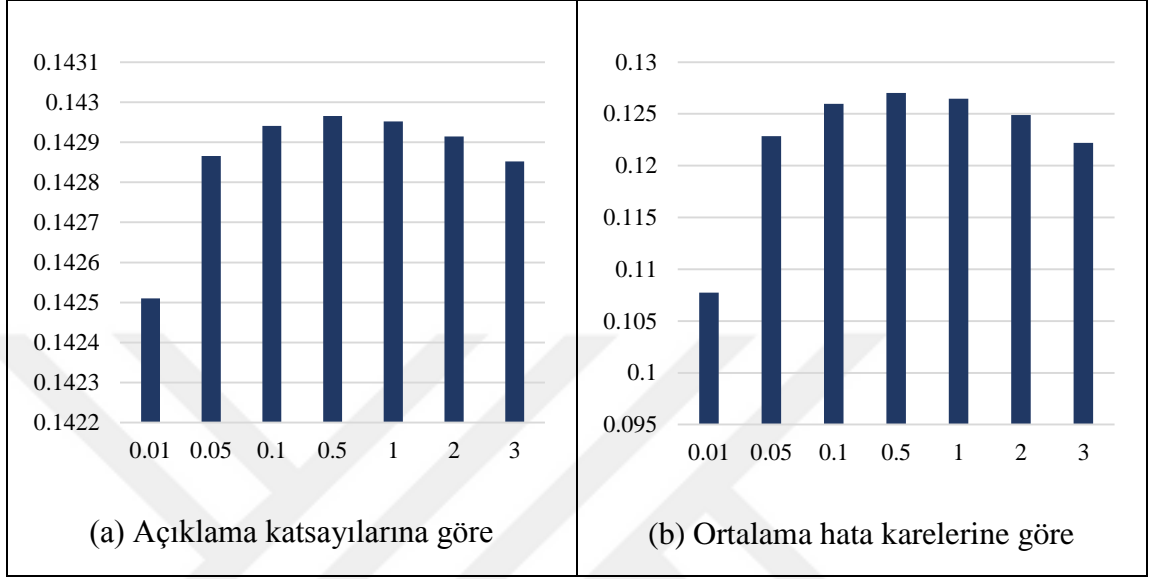
Şekil 6.10. Gaussian çekirdek fonksiyonu için farklı genişlik değerleri ile elde edilen sonuçlar

Farklı genişlik değerlerine sahip Gaussian çekirdek fonksiyonlarının en iyi tahmin performansını verecek kombinasyonunun bulunması için negatif olmayan doğrusal kombinasyon (konik kombinasyon) yaklaşımı uygulanacaktır. Bu yaklaşımın uygulanmasının sebebi pozitif katsayılar sayesinde farklı çekirdek fonksiyonlarının göreceli önem dereceleri hakkında yorum yapılabilmesine olanak tanınmasıdır.

$$K = \left\{ \sum_{m=1}^P \eta_m \mathbf{K}_m, \eta_m \geq 0 \right\} \quad (6.8)$$

Farklı genişlik değerlerine sahip Gaussian çekirdek fonksiyonlarına atanacak ağırlıkların bulunmasında ilgili modellerle elde edilen ortalama hata karesi ve açıklama katsayısı değerlerinden faydalanılmıştır. En iyi tahmin performansına ulaşılabilmesi için ortalama

hata karesi küçük ve açıklama katsayısı büyük olan çekirdek fonksiyonunun ağırlığının daha yüksek olması amaçlanmaktadır. Elde edilen ağırlıklar Şekil 6.11’de sunulmuştur.



Şekil 6.11. Farklı genişlikte Gaussian çekirdek fonksiyonlarına atanan ağırlıklar

Şekil 6.11’den anlaşılacağı üzere açıklama katsayısı büyük ve ortalama hata karesi küçük olan çekirdek fonksiyonuna en yüksek ağırlık atanmıştır. Diğer parametre değerlerine sahip fonksiyonlar da performansları ile doğru orantılı ağırlıklar ile modelde yer almışlardır.

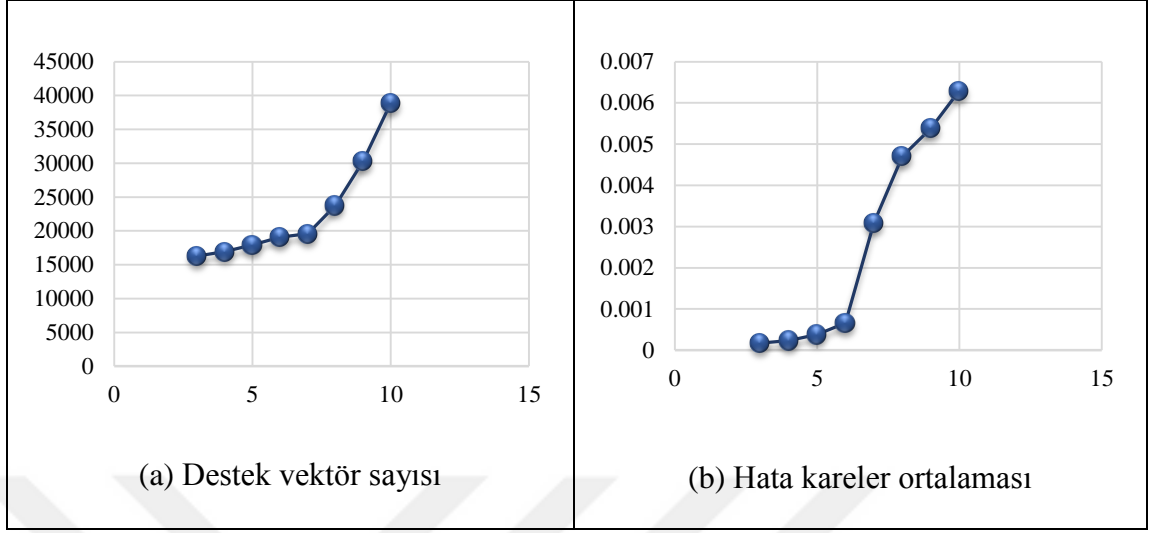
6.4.2. Polinomial Çekirdek Fonksiyonlarının Konik Kombinasyonu

Bu kısımda farklı dereceden Polinomial çekirdek fonksiyonlarının konik kombinasyonu oluşturulacaktır.

Çizelge 6.8. Polinomial çekirdek fonksiyonu için farklı parametre değerleri ile elde edilen sonuçlar ($\varepsilon = 0,01$ ve $C = 10$)

Parametre değeri	Destek Vektör sayısı	HKO
3	16269	0,000173381
4	16931	0,000237667
5	17948	0,000381014
6	19078	0,000663283
7	19590	0,00309364
8	23760	0,00470943
9	30309	0,00537969
10	38864	0,00629905

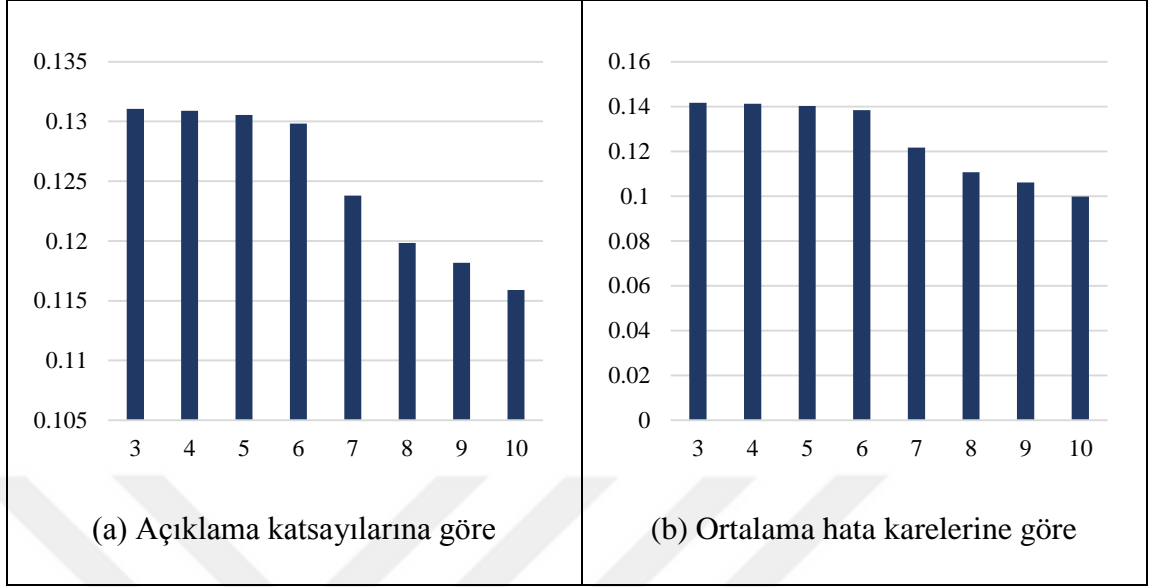
Çizelge 6.8'den anlaşıldığı üzere Polinomial çekirdek fonksiyonunun derecesi büyüdükçe destek vektör sayısı artmakta, diğer bir deyişle model kompleksliği ve eğitim süresi artmaktadır. Bununla birlikte hata kareler ortalamasında artış görülmekte, dolayısıyla fonksiyon derecesindeki artışın model performansında düşüşe sebep olduğu çıkarımı yapılmaktadır. Sonuç olarak Polinomial çekirdek fonksiyonunun derecesinin optimum değeri 3 olarak bulunmuştur. Elde edilen bilgiler Şekil 6.12'de özetlenmiştir.



Şekil 6.12. Polinomial çekirdek fonksiyonu için farklı parametre değerleri ile elde edilen sonuçlar

Farklı dereceden Polinomial çekirdek fonksiyonlarının en iyi tahmin performansını verecek kombinasyonunun bulunması için de negatif olmayan doğrusal kombinasyon (konik kombinasyon) yaklaşımı uygulanacaktır.

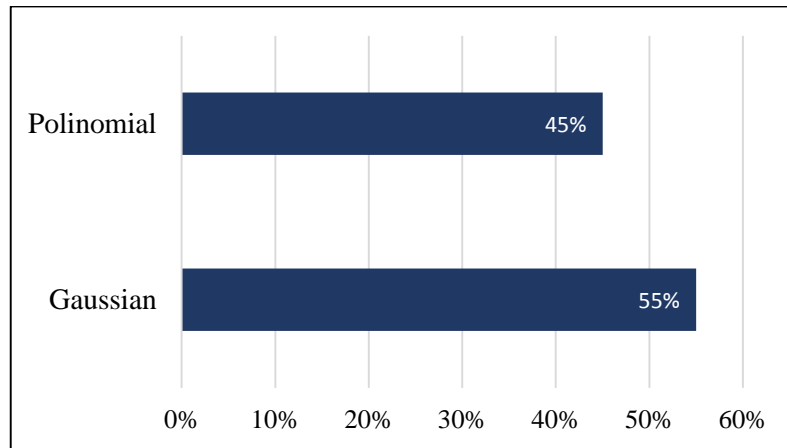
Farklı dereceden Polinomial çekirdek fonksiyonlarına atanacak ağırlıkların bulunmasında benzer şekilde ilgili modellerle elde edilen ortalama hata karesi ve açıklama katsayısı değerlerinden faydalanılmıştır. En iyi tahmin performansına ulaşılabilmesi için ortalama hata karesi küçük ve açıklama katsayısı büyük olan çekirdek fonksiyonunun ağırlığı daha yüksek olacaktır. Elde edilen ağırlıklar Şekil 6.13'te sunulmuştur.



Şekil 6.13. Farklı dereceli Polinomial çekirdek fonksiyonlarına atanan ağırlıklar

6.4.3. Farklı Çekirdek Fonksiyonlarının Konveks Kombinasyonu

Gaussian ve Polinomial çekirdek fonksiyonları yaklaşık olarak aynı performansı gösterdiklerinden iki çekirdek fonksiyonunun birleştirilmesinin faydalı olacağı değerlendirilmiştir. Bu durumda çekirdek fonksiyonlarının kombinasyonu $\eta_m \in \mathbb{R}_+^P$ ve $\sum_{m=1}^P \eta_m = 1$ olmak üzere ağırlıklı ortalamaları olarak alınmaktadır ve iki çekirdek fonksiyonu birbirini tamamlayıcı özelliktedir ($\eta_2 = 1 - \eta_1$). Bulunan ağırlıklar Şekil 6.14'de gösterilmektedir.



Şekil 6.14. Gaussian ve Polinomial çekirdek fonksiyonlarına atanan ağırlıklar

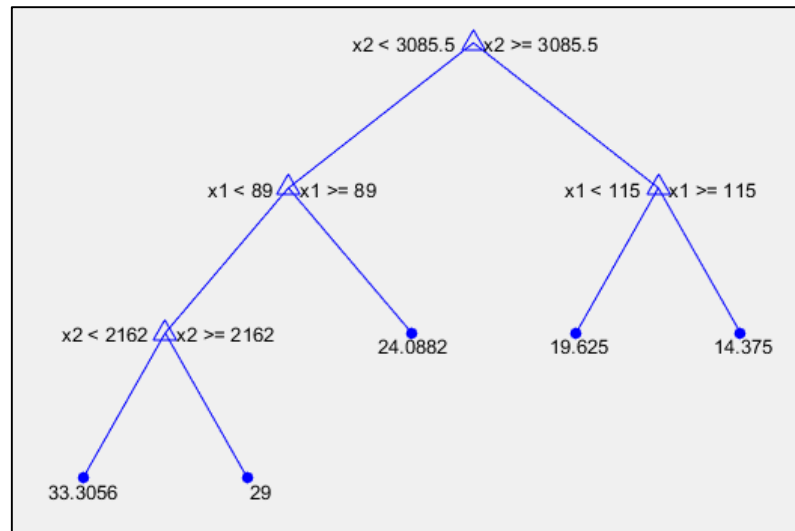
6.5. DİĞER REGRESYON YÖNTEMLERİ İLE KARŞILAŞTIRMA

Bu kısımda regresyon problemlerinin çözümünde kullanılan diğer makine öğrenme yöntemleri hakkında özet bilgi verilecek ve aynı veri seti üzerinde uygulamaları sunulacaktır. Modeller arasındaki ilişkilerin, farklılıkların, birbirlerine olan üstünlüklerinin incelenmesi ve elde edilen sonuçların karşılaştırılması amaçlanmaktadır. Modellerin uygulanmasında MATLAB programı kullanılmıştır.

6.5.1. Regresyon Ağaçları

Karar ağaçları (Breiman ve diğ., 1984), sınıflandırma (sınıflandırma ağaçları) ve regresyon (regresyon ağaçları) problemlerinin çözümünde veriye bağlı olarak yanıt değişkeninin tahmin edilmesi amacıyla kullanılırlar. Başlangıç (kök) düğüm noktasından son yaprak noktasına kadar ağaçtaki kararlar takip edilir. Son yaprak noktası yanıtı içermektedir. Sınıflandırma ağaçları nominal, regresyon ağaçları ise sayısal yanıt değişkenlerini vermektedir. Regresyon ağacı modelinin çözümünde MATLAB Statistics and Machine Learning Toolbox kullanılmıştır.

Ağaç yapısı ikilidir. Tahmin sürecindeki her adımda bir açıklayıcı değişkenin değeri incelenmektedir. Şekil 6.15'te basit bir regresyon ağacı modeli örnek olarak sunulmuştur.



Şekil 6.15. Örnek regresyon ağacı modeli (MATLAB Regression tree viewer)

Regresyon ağaçları yönteminde modelin oluşturulmasında aşağıdaki adımlar uygulanmaktadır (Breiman ve diğ., 1984):

i. Tüm girdi verileri ile başlanır ve her tahmin edici üzerindeki olası tüm ikili ayrılmalar incelenir.

ii. En iyi optimizasyon kriterine sahip olan ayırıcı seçilir. Bir ayırma sonucu elde edilen çocuk düğüm noktası minimum yaprak boyutu parametre değerinden daha az sayıda gözleme sahip olabilir. Bunu önlemek için model minimum yaprak boyutu kriterine bağlı olarak en iyi optimizasyon kriterine sahip olan ayırıcıyı seçmektedir.

iii. Ayırma işlemi uygulanır.

iv. Bu adımlar elde edilen iki çocuk düğüm noktası için tekrarlanır.

Ayırma işlemi aşağıdaki durumlardan birisi gerçekleştiğinde durdurulur:

i. Bu düğüm noktasında gözlenen yanıt değişkeni için hata kareler ortalaması, tüm verideki gözlenen yanıt değişkeni için hata kareler ortalaması ile her bir düğüm noktası için kuadratik hata tolerans parametresi değerinin çarpımından küçüktür.

ii. Bu düğüm noktasında minimum ebeveyn büyüklüğünden daha az sayıda gözlem mevcuttur.

iii. Bu düğüm noktasında uygulanacak herhangi bir ayırma işlemi minimum yaprak büyüklüğünden daha az sayıda gözleme sahip çocuk düğüm noktaları vermektedir.

iv. Algoritma maksimum ayırma sayısı kadar düğüm noktası için ayırma işlemini uygulamıştır.

Optimizasyon kriteri hata kareler ortalamasıdır. Eğitim verisi için elde edilen tahminlerin hata kareler ortalamasını minimum yapacak olan ayırma seçilir.

Bu aşamada regresyon ağaçlarının performansının değerlendirilmesi ve ağaç büyüklüğünün belirlenmesi konularına değinilecektir.

Eğitim verisindeki yanıt değişkeni değerleri ile eğitim verisindeki girdi değişkenlerine bağlı olarak ağaç modeli ile elde edilen tahmin değerleri arasındaki fark incelenir. Eğer fark büyük ise tahmin performansının iyi olmadığı değerlendirilir. Ancak farkın küçük olması yeni veri seti için başarılı tahminler elde edileceğini garanti etmez.

Ağacın tahmin doğruluğunun daha kapsamlı analizi için çapraz geçerlilik yöntemi uygulanır. Çapraz geçerlilik yönteminde eğitim verisi rasgele, örneğin 10 parçaya, ayrılır. 10 yeni ağaç kalan diğer 9 parça üzerinde eğitilir. Sonrasında yeni ağaçların her birinin eğitim setinde bulunmayan veri üzerindeki tahmin doğruluğu incelenir. Bu yöntem elde edilen ağaç modelinin tahmin doğruluğunun analizi için güvenilir sonuçlar vermektedir, çünkü yeni ağaçlar yeni veriler üzerinde test edilmektedir.

Karar ağacının büyüklüğü belirlenirken model basitliği ve tahmin gücü göz önünde bulundurulmaktadır. Çok sayıda yaprağa sahip kompleks bir ağaç modeli genellikle eğitim verisi üzerinde yüksek doğruluk dereceli tahminler vermektedir. Ancak bu durum ağaç modelinin bağımsız test verisi üzerinde de aynı şekilde başarılı sonuçlar vereceğini göstermemektedir. Çok yapraklı bir ağaç modeli aşırı uyum sorununa meyillidir ve test verisi üzerindeki doğruluk derecesi genellikle eğitim verisine göre çok daha düşük olmaktadır. Bunun aksine küçük büyüklükte bir ağaç modeli eğitim verisi üzerinde yüksek doğruluk derecesine ulaşmaz. Ancak küçük büyüklükteki bir ağaç daha stabildir, eğitim ve test verileri üzerindeki doğruluk dereceleri birbirine yakın olur. Ayrıca yorumlanması da daha kolaydır.

Elde edilen karar ağacı modelinin derinliğinin kontrol edilmesinde kullanılan 3 parametre mevcuttur:

i. Maksimum ayırma sayısı ağaç için dal düğüm noktalarının maksimum ayırma sayısıdır. Derin bir ağaç modeli elde etmek için bu parametreye büyük bir değer verilmelidir, varsayılan değer $[boyut(X, 1)]$ 'dir.

ii. Minimum yaprak büyüklüğü her yaprağın sahip olması istenen minimum gözlem sayısıdır. Derin bir ağaç modeli elde etmek için bu parametreye küçük bir değer verilmelidir, varsayılan değer 1'dir.

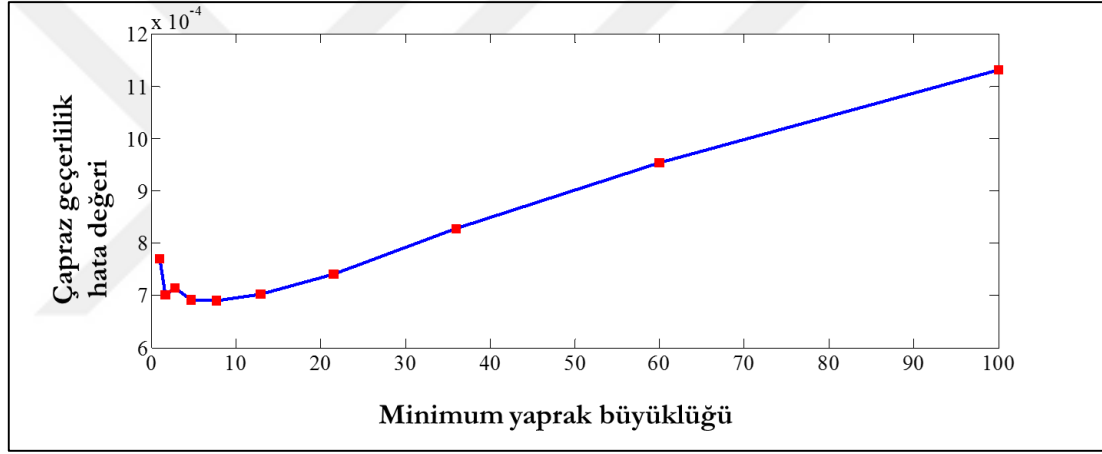
iii. Minimum ebeveyn büyüklüğü her dal düğüm noktasının sahip olması istenen minimum gözlem sayısıdır. Derin bir ağaç modeli elde etmek için bu parametreye küçük bir değer verilmelidir, varsayılan değer 10'dur.

Varsayılan parametre değerleri kullanılarak oluşturulan model ile elde edilen sonuçlar Çizelge 6.9'da sunulmuştur.

Çizelge 6.9. Regresyon ağacı modeli ile elde edilen sonuçlar

Eğitim veri seti üzerinde hata kareler ortalaması	0,00026
Çapraz geçerlilik hata kareler ortalaması	0,00076

Karar ağacının derinliğinin kontrol edilmesi ve uygun ağaç büyüklüğünün seçilmesi için çapraz geçerlilik yönteminden faydalanılmaktadır. Regresyon ağacı için farklı minimum yaprak büyüklüğü değerleri kullanılarak çapraz geçerlilik hatası regresyon ağaçları oluşturulmuştur. Elde edilen sonuçlar Şekil 6.16’da sunulmuştur. En iyi yaprak büyüklük değeri her bir yaprak için 5 gözlem olarak bulunmuştur.



Şekil 6.16. Farklı minimum yaprak büyüklükleri için çapraz geçerlilik hata değerleri

Her bir yaprak için en az 5 gözlem içeren optimum ağaç modeli ile varsayılan parametre değerlerini kullanan ve her bir ebeveyn düğüm noktası için 10 ve her bir yaprak için 1 gözlem içeren modellerin uygulanması ile elde edilen sonuçlara ait karşılaştırma Çizelge 6.10’da sunulmuştur.

Çizelge 6.10. Temel model ve optimum model sonuçlarının karşılaştırılması

	Temel model	Optimum model
Eğitim veri seti üzerinde hata kareler ortalaması	0,00026	0,00034
Çapraz geçerlilik hata kareler ortalaması	0,00076	0,00070

Çizelge 6.10'da görüldüğü üzere optimum ağaç modeli eğitim verisi üzerinde daha yüksek hata değerine sahip olmasına rağmen yeni veri seti üzerinde test edildiğinde daha yüksek doğruluk dereceli tahmin performansı göstermektedir.

Eğitilen regresyon ağacı modelinin test verisi üzerinde uygulanması sonucunda hata kareler ortalaması değeri 0.0019 olarak bulunmuştur. Normalleştirilmiş hata kareler ortalamasının karekökü ise 0.0785 olarak hesaplanmıştır.

Karar ağaçları yönteminde kurulan regresyon ağaç modeli için açıklayıcı değişkenlerin önem derecelerinin tahmininin hesaplanması şu şekilde yapılmaktadır: Her bir açıklayıcı değişkendeki ayrılmalar sonucunda hata kareler ortalamasında meydana gelen değişim miktarları toplanır ve elde edilen toplam dal noktası sayısına bölünür.

Eğer ağaç vekil ayrımlar olmaksızın oluşturulmuş ise bu toplam her dal noktasındaki en iyi ayrımları kapsamaktadır. Eğer ağaç vekil ayrımlar kullanılarak oluşturulmuş ise toplam her dal noktasındaki en iyi ayrımları ve vekil ayrımları kapsamaktadır. Her bir noktada hata kareler ortalaması ayırma noktası olasılığı ile ağırlıklandırılmış ayırma noktası hatası olarak hesaplanmaktadır. Bu ayırmaya ilişkin değişken önem derecesi ebeveyn noktasının hata kareler ortalaması ile iki çocuk için hesaplanan hata kareler ortalaması değerlerinin toplamının farkı alınarak bulunmektedir.

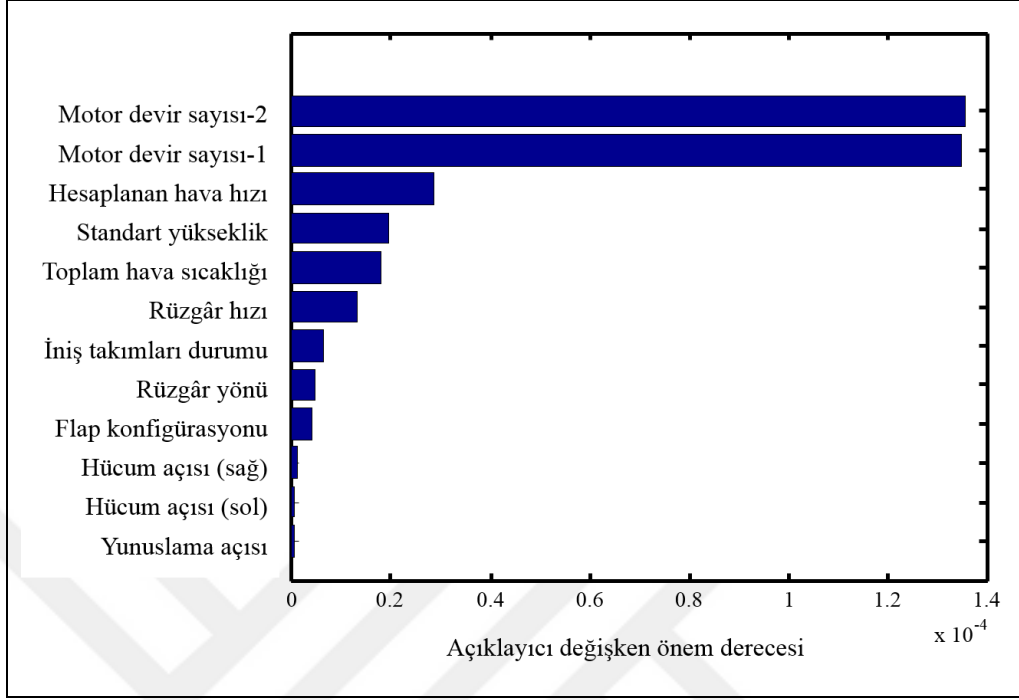
Vekil ayrımlar kullanılarak oluşturulan ağaçta açıklayıcı değişkenlerin önem dereceleri değişken sırasına bağlı olmamaktadır, ancak vekil ayrımlar kullanılmadığında değişkenlerin önemleri sıralarına bağlı olmaktadır. Vekil ayrımlar kullanılarak oluşturulan ağaçta yaprak birleştirme veya budama işlemlerinden önce açıklayıcı değişkenlerin tahmini önem dereceleri hesaplanmaktadır. Vekil ayrımlar kullanılmadığında birleştirme veya budama işlemlerinden sonra açıklayıcı değişkenlerin tahmini önem dereceleri hesaplanmaktadır. Dolayısıyla ağacın yaprakların birleştirilmesi veya budama işlemi ile küçültülmesi vekil ayırma kullanılmadan oluşturulan ağaç modeli için hesaplanan açıklayıcı değişken önem derecelerini etkilemekte, vekil ayırma kullanılarak oluşturulan ağaç modeli için hesaplanan açıklayıcı değişken önem derecelerini etkilememektedir.

Regresyon ağacı modeli için vekil ayırma yönteminin kullanıldığı ve kullanılmadığı durumda hesaplanan açıklayıcı değişken önem dereceleri Çizelge 6.11'de sunulmuştur.

Çizelge 6.11. Regresyon ağacı modeli için açıklayıcı değişkenlerin önem dereceleri

Nu.	Değişken adı	Önem derecesi	
		Vekil ayırma yok	Vekil ayırma var
1	Toplam hava sıcaklığı	0,0003	0,0179
2	Motor devir sayısı-1	0,0002	0,1351
3	Motor devir sayısı-2	0,1462	0,1356
4	Standart yükseklik	0,0092	0,0194
5	Hücum açısı (sol)	0,0000	0,0005
6	Hücum açısı (sağ)	0,0000	0,0011
7	Yunuslama açısı	0,0000	0,0004
8	Hesaplanan hava hızı	0,0002	0,0285
9	Rüzgâr hızı	0,0000	0,0130
10	Rüzgâr yönü	0,0000	0,0046
11	Flap konfigürasyonu	0,0000	0,0040
12	İniş takımları durumu	0,0000	0,0064

Değişken sırasına bağlı olmaksızın, vekil ayırma kullanılarak, elde edilen önem dereceleri Şekil 6.17’de sıralanmıştır.



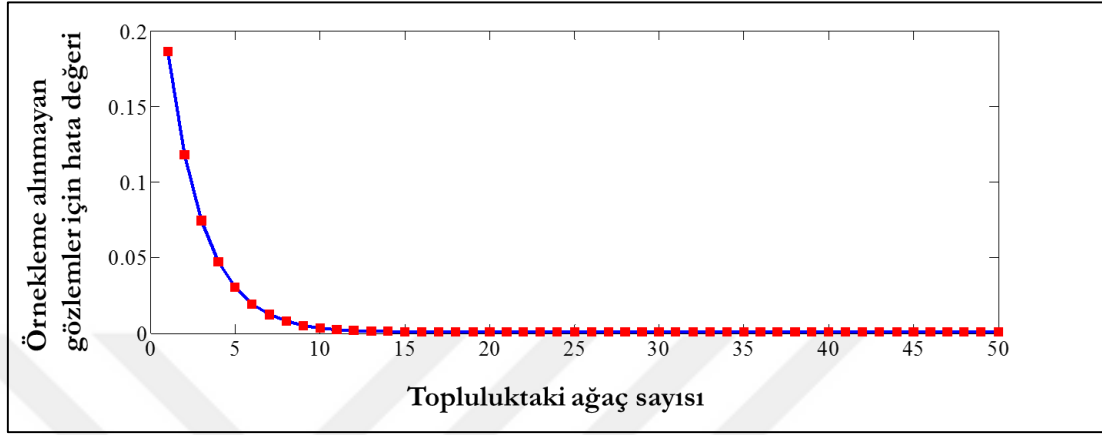
Şekil 6.17. Açıklayıcı değişkenlerin önem dereceleri

Bulunan önem dereceleri incelendiğinde yakıt tüketimi modeli üzerinde en etkili olan değişkenlerin motor devir sayısına ilişkin değişkenler olduğu görülmektedir.

Regresyon ağaçlarında tahmin performansının artırılmasında kullanılan yöntemlerden birisi karar ağaçları topluluğu (ensemble) için bootstrap örnekleme (bootstrap aggregation) yöntemidir (Breiman, 1996). Bu yöntemde topluluktaki her ağaç girdi verisinden bağımsız olarak çekilmiş bootstrap örneklemeden oluşturulmaktadır. Ağaç topluluğunun yeni veri için tahmin değeri bireysel ağaç modellerinden elde edilen tahmin değerlerinin ortalaması hesaplanarak bulunmaktadır. Topluluğun tahmin hatası ise şu şekilde hesaplanabilir: Her bir ağacın bootstrap örneklemesine alınmayan (out-of-bag) gözlemler için tahmin değeri bulunur. Her bir gözlem için topluluktaki tüm ağaçlar tarafından bulunan tahmin değerlerinin ortalaması hesaplanır. Bootstrap örneklemesi dışında kalan yanıt değişkenlerinin tahmini değerleri ile bu gözlemlere ait gerçek değerler karşılaştırılır.

Oluşturulan regresyon ağacı sayısına karşılık örnekleme dışında kalan gözlemler için bulunan hata değerleri Şekil 6.18’de sunulmuştur. Tek regresyon ağacı ile oluşturulan modele göre tahmin performansında iyileşme olduğu görülmektedir. 50 adet regresyon

ağaçından oluşan topluluğun hata kareler ortalaması değeri 0.0014 olarak bulunmuştur. Normalleştirilmiş hata kareler ortalamasının karekökü ise 0.0689 olarak hesaplanmıştır.



Şekil 6.18. Regresyon ağaçları toplulukları ve hata değerleri

6.5.2. Lasso Düzleştirme

Genelleştirilmiş doğrusal modellerde açıklayıcı değişken sayısının azaltılması ve önemli değişkenlerin belirlenmesi amacıyla Lasso düzleştirme (Lasso regularization) yöntemi kullanılmaktadır (Tibshirani, 1996; Friedman ve diğ., 2010). Lasso, katsayıların tahmini değerlerini kısıtlayan bir ceza terimi içermektedir. Bu bakımdan Ridge regresyon yöntemine benzemektedir. Ridge regresyon yönteminden farklı olarak ceza terimi arttıkça Lasso yönteminde daha çok katsayı değeri sıfır olmaktadır. Diğer bir deyişle Lasso yöntemi ile daha az sayıda açıklayıcı değişkene sahip olan daha basit bir model elde edilmektedir. Bu bakımdan adımsal regresyon veya diğer model seçimi ve boyut indirgeme yöntemlerine alternatif olarak uygulanmaktadır. Elastik net de benzer bir yöntem olmakla birlikte yüksek ilişkili değişkenlerin söz konusu olduğu durumlarda daha başarılı sonuçlar vermektedir.

Negatif olmayan λ değerleri için Lasso aşağıda verilen problemi çözmektedir (Tibshirani, 1996):

$$\min_{\beta_0, \beta} \frac{1}{N} \text{Sapma}(\beta_0, \beta) + \lambda \sum_{j=1}^p |\beta_j| \quad (6.9)$$

Burada,

i. Sapma, β_0 sabit katsayı ve β açıklayıcı değişken katsayıları kullanıldığında model uyumunun yanıt değerlerinden olan sapmasıdır. λ -cezalandırılmış sapmanın minimize edilmesi λ -cezalandırılmış log olabilirliğin maksimize edilmesine eşdeğerdir.

ii. N gözlem sayısıdır.

iii. λ negatif olmayan düzleştirme (regularization) parametresidir. λ değeri arttıkça β katsayılarının sıfıra eşit olmayan eleman sayısı azalmaktadır.

Elastik net algoritmasından farklı olarak Lasso problemi L^1 normu içermektedir. α değeri 0 ile 1 arasında olacak şekilde belirlendiğinde negatif olmayan λ değerleri için Elastik net aşağıda verilen problemi çözmektedir (Tibshirani, 1996):

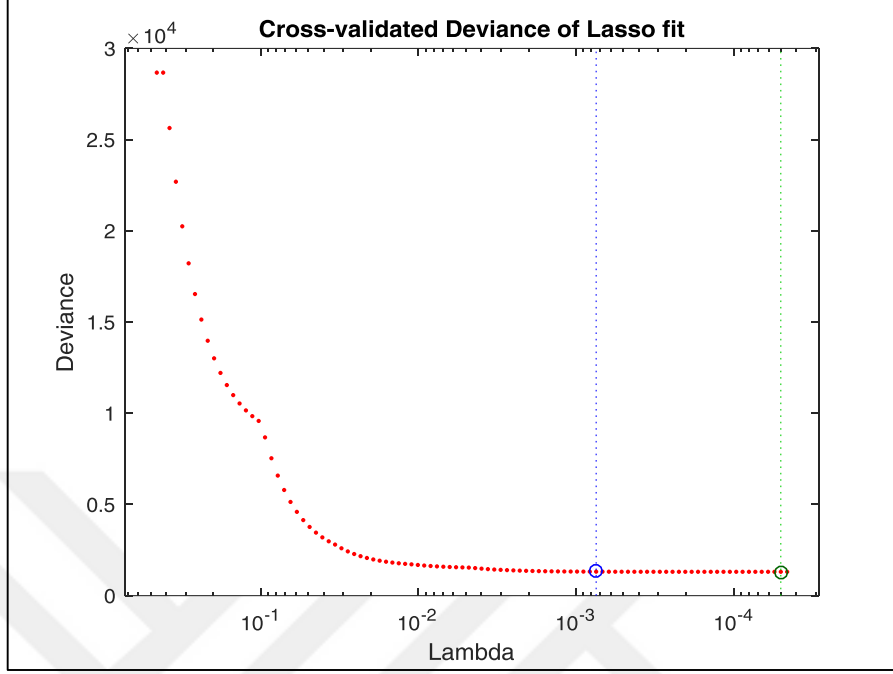
$$\min_{\beta_0, \beta} \left(\frac{1}{N} \text{Deviance}(\beta_0, \beta) + \lambda P_\alpha \sum_{j=1}^p (\beta) \right) \quad (6.10)$$

Burada,

$$P_\alpha(\beta) = \frac{(1 - \alpha)}{2} \|\beta\|_2^2 + \alpha \|\beta\|_1 = \sum_{j=1}^p \left(\frac{(1 - \alpha)}{2} \beta_j^2 + |\beta_j| \right) \quad (6.11)$$

$\alpha = 1$ durumunda Elastik net Lasso ile aynı olmaktadır. Diğer α değerleri için ceza terimi $P_\alpha(\beta)$, L^1 norm ile L^2 norm arasında olmaktadır. α değeri 0'a yaklaştıkça Elastik net Ridge regresyona yaklaşmaktadır (Tibshirani, 1996).

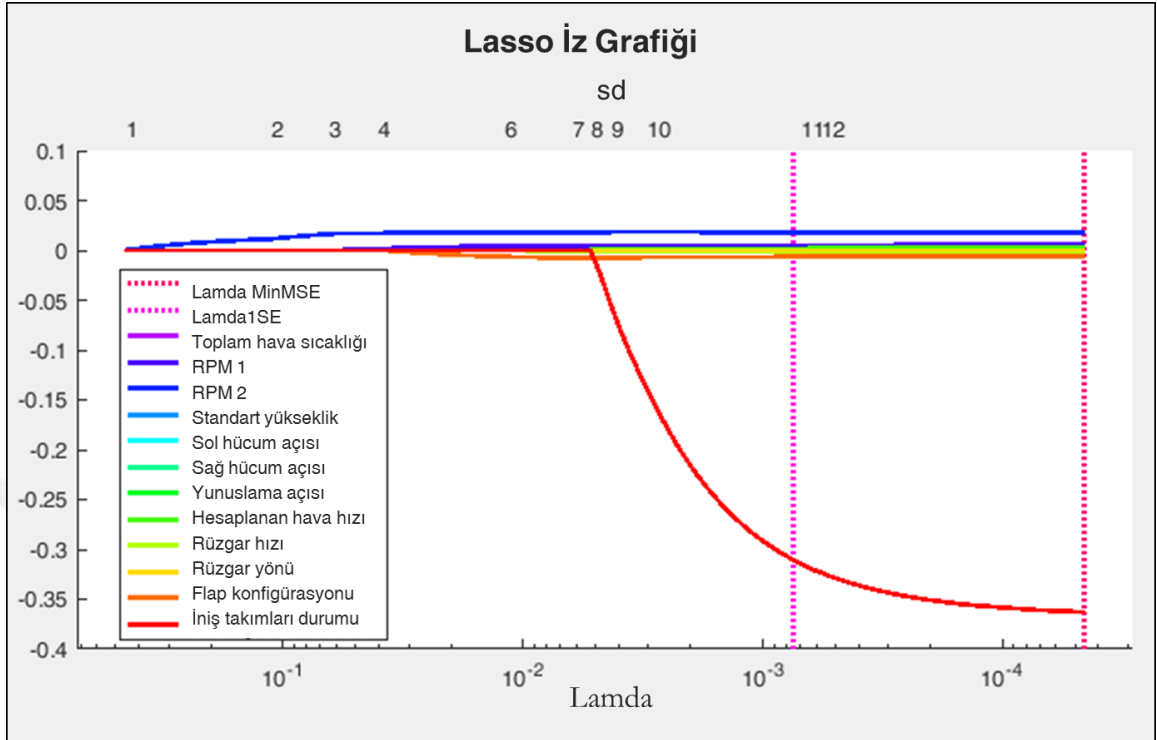
λ düzleştirme parametresinin etkisini göstermek amacıyla çapraz geçerlilik grafiği Şekil 6.19'da sunulmuştur.



Şekil 6.19. Çapraz geçerlilik grafiği

Yeşil çember ve kesikli çizgi minimum çapraz geçerlilik hatasına sahip λ değerini göstermektedir. Mavi çember ve kesikli çizgi ise minimum çapraz geçerlilik hatası artı bir standart sapma noktasını göstermektedir.

Lasso düzleştirme yönteminin uygulanması ile elde edilen iz grafiği (trace plot) Şekil 6.20'de sunulmuştur. Minimum çapraz geçerlilik hatasına sahip λ değerinde tüm açıklayıcı değişkenler önemli bulunmuştur. Minimum çapraz geçerlilik hatası artı bir standart sapma noktasında ise sol hücum açısı (angle of attack) hariç diğer açıklayıcı değişken önemli bulunmuştur.



Şekil 6.20. Lasso iz grafiği

Karşılaştırma maksadıyla doğrusal regresyon modeli ile elde edilen sonuçlar Şekil 6.21’de sunulmuştur.

```

Linear regression model:
  y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 + x11 + x12

Estimated Coefficients:

```

	Estimate	SE	tStat	pValue
x1	-0.0037426	1.4985e-05	-249.76	0
x2	0.012603	0.00017657	71.376	0
x3	0.0097849	0.00017645	55.455	0
x4	-3.9847e-05	9.1731e-08	-434.39	0
x5	-0.00022168	2.243e-05	-9.883	5.0587e-23
x6	3.2184e-06	2.3049e-05	0.13964	0.88895
x7	5.8227e-05	9.0887e-06	6.4065	1.4955e-10
x8	0.0013751	1.7272e-05	79.617	0
x9	-0.00056858	6.2023e-05	-9.1672	4.9442e-20
x10	-3.5661e-05	2.9568e-06	-12.061	1.8034e-33
x11	-0.0096387	6.7817e-05	-142.13	0
x12	-0.34491	0.0034819	-99.059	0

```

Number of observations: 97536, Error degrees of freedom: 97524
Root Mean Squared Error: 0.125

```

Şekil 6.21. Doğrusal regresyon modeli elde edilen sonuçlar

Düzleştirme (regularization) uygulandığı ve uygulanmadığı modeller sonucunda hesaplanan ortalama hata kare değerleri Çizelge 6.12’de sunulmuştur.

Çizelge 6.12. Lasso ve doğrusal regresyon model sonuçlarının karşılaştırılması

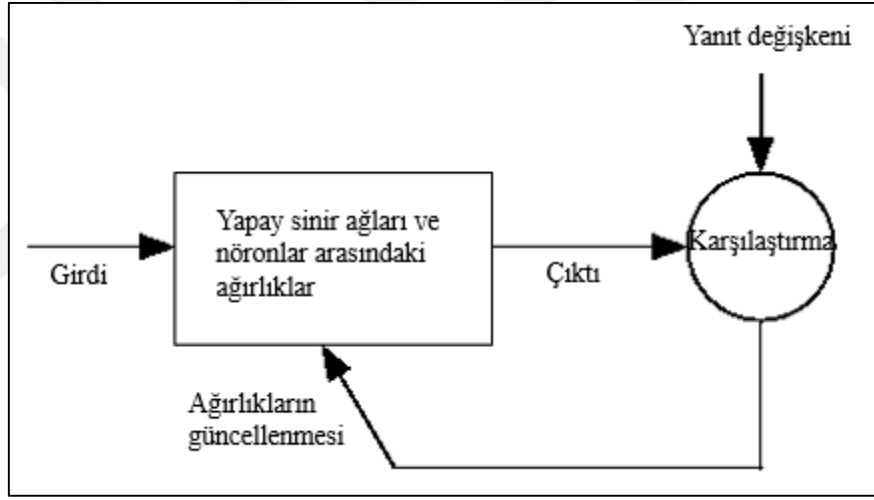
Lasso HKO	0.015713
Tam model HKO	0.013468

Lasso yöntemi ile önemli bulunan açıklayıcı değişkenler kullanılarak oluşturulan modelin hata değeri tüm değişkenlerin kullanıldığı doğrusal modele yakındır. Lasso yöntemi genellikle aşırı uyumun önlenmesi ve gereksiz değişkenlerin model dışında bırakılması amacıyla kullanılmaktadır.

Lasso yöntemi ile oluşturulan modelin doğruluk derecesi istenen seviyede olmamasına rağmen değişken seçiminin çözülmek istenen probleme ve eldeki veri setine uygun olduğunun doğrulanması bakımından destekleyici niteliktedir.

6.5.3. Yapay Sinir Ağları

Yapay sinir ağları (Nabney, 2001) paralel faaliyet gösteren unsurlardan oluşmaktadır. Bu unsurlar biyolojik sinir sistemlerinden esinlenerek oluşturulmuştur ve unsurlar arasındaki bağlantılar ağ fonksiyonunu büyük ölçüde belirlemektedir. Yapay sinir ağı modeli belirli bir fonksiyonu uygulamak için unsurlar arasındaki bağlantıların değerleri (ağırlıkları) güncellenerek eğitilebilmektedir. Genel olarak yapay sinir ağları model tarafından tahmin edilen çıktı değerleri ve gerçek çıktı değerlerinin karşılaştırılmasına dayanılarak eğitilir. Bu süreç Şekil 6.22’de özetlenmiştir. Yapay sinir ağları modeli regresyon, sınıflandırma ve kümeleme problemlerinin çözümünde kullanılmaktadır.

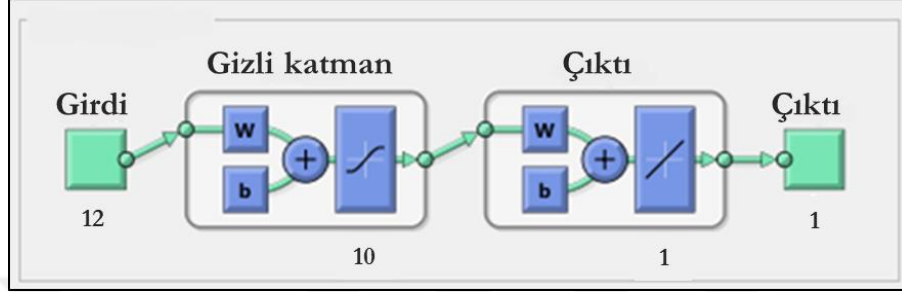


Şekil 6.22. Yapay sinir ağı modelinin eğitilmesi

Modelin uygulanmasında MATLAB Neural Network Toolbox kullanılmıştır. Geçerlilik ve test veri setlerinin her biri orijinal veri setinin %15’i olarak belirlenmiştir. Eğitim seti eğitim aşamasında ağa sunulmuştur ve ağ eğitim setindeki hatalar temel alınarak güncellenmiştir. Geçerlilik seti ağın genelleştirme performansının değerlendirilmesinde ve aşırı uyum sorununun önlenmesi amacıyla genelleştirme performansındaki iyileşme durduğunda eğitimin sonlandırılması için kullanılmıştır. Test verisi eğitim aşamasında kullanılmadığından eğitim süresince ve sonrasında ağ performansının bağımsız bir ölçümünü sunmaktadır.

Regresyon probleminin çözümü için standart modelde iki katmanlı ileri beslemeli bir ağ, gizli katmanda sigmoid transfer fonksiyonu ve çıktı katmanında doğrusal transfer

fonksiyonu kullanılmıştır. Gizli nöron sayısının başlangıç değeri 10 olarak alınmıştır. Ağın eğitim performansına göre bu sayı arttırılabilmektedir. Kullanılan ağ modeli Şekil 6.23'te sunulmuştur.

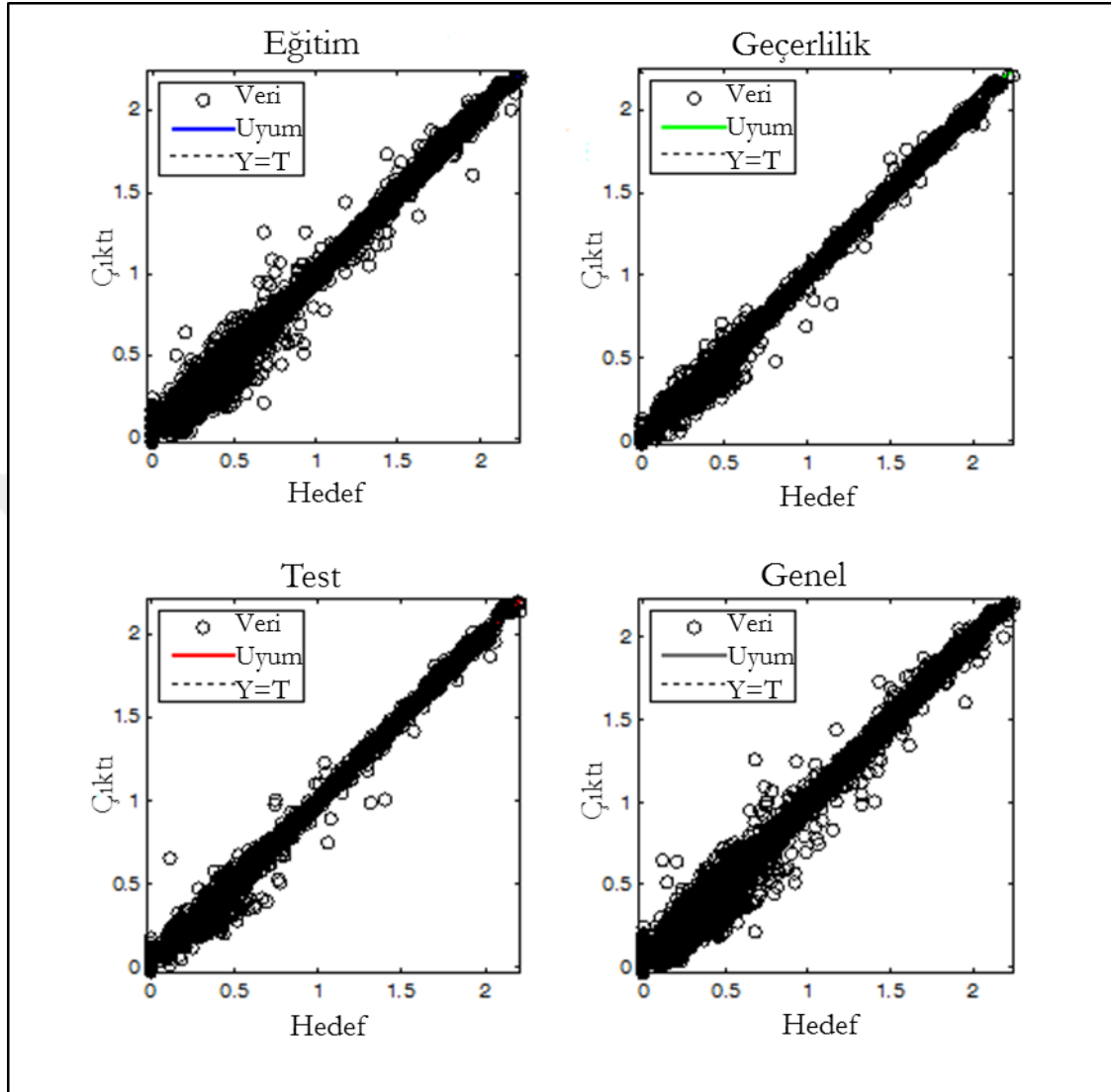


Şekil 6.23. Yapay sinir ağı modeli

Ağın eğitilmesi için Levenberg-Marquardt (LM) algoritması kullanılmıştır. Bu algoritma bellekte daha fazla yer kaplamasına rağmen eğitim süresinin daha kısa olması sebebiyle tercih edilmektedir.

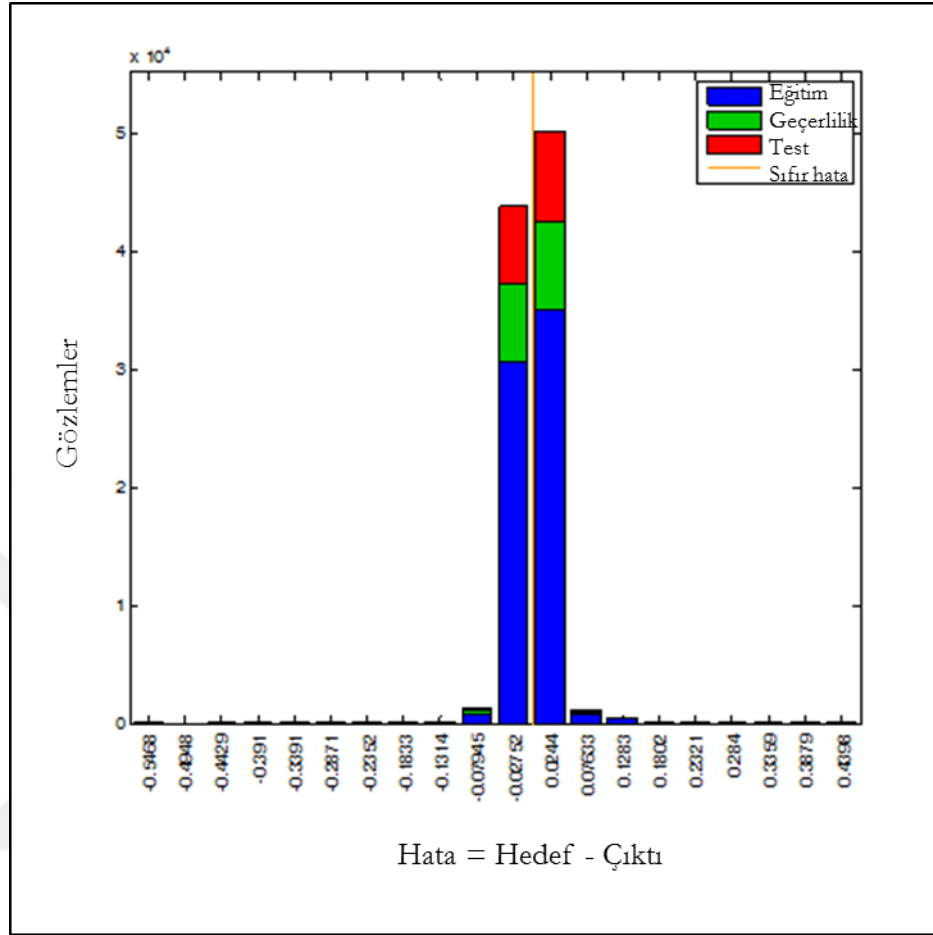
Genelleştirme performansındaki iyileşme durduğunda, diğer bir deyişle geçerlilik veri setinin hata kareler ortalamasında artış görüldüğü noktada, eğitim durdurulur. Burada şunu belirtmek gerekir ki ağın birden fazla defa eğitilmesi, farklı başlangıç değerleri ve örnekleme sebebiyle farklı sonuçlar verebilmektedir.

Şekil 6.24'te sunulan regresyon grafikleri eğitim, geçerlilik ve test verisindeki hedef değerlerine ilişkin ağ çıktılarını göstermektedir. Mükemmel uyum için, verilerin 45 derece doğrusu boyunca yer alması istenir. Bu problem için uyumun tüm veri setleri için iyi olduğu söylenebilir. Eğer daha yüksek doğruluk dereceli bir model oluşturulmak istenirse ağ yeniden eğitilebilir. Bu ağdaki başlangıç ağırlık ve yan değerlerini değiştirecektir ve yeniden eğitim sonucunda daha yüksek performanslı bir model elde edilebilir.



Şekil 6.24. Regresyon grafikleri

Modelin performansını değerlendirmek için başvurulan ilave bir yöntem ise hataların histogram grafiğinin incelenmesidir. Şekil 6.25'te mavi barlar eğitim verisini, yeşil barlar geçerlilik verisini ve kırmızı barlar da test verisini göstermektedir. İncelenen grafikte hataların çoğunun -0.1 ve +0.1 arasında yer aldığı görülmektedir.



Şekil 6.25. Hataların histogram grafiği

Elde edilen sonuçlar Şekil 6.26'da özetlenmiştir.

	Örnekler	HKO
Eğitim	68276	0,0005858
Geçerlilik	14630	0,0005677
Test	14630	0,0005931

Şekil 6.26. Yapay sinir ağı modeli ile elde edilen sonuçlar

Oluşturulan modelin test verisi üzerinde uygulanması sonucunda hata kareler ortalaması değeri 0.000714 olarak hesaplanmıştır.

6.5.4. Gaussian Süreç Regresyon

Gaussian süreç (Gaussian process) regresyon modelleri nonparametrik çekirdek-tabanlı olasılıksal modellerdir. Eğitim seti $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ şeklinde verilsin. Burada $\mathbf{x}_i \in \mathbb{R}^D$ D boyutlu açıklayıcı değişkenler ve $y_i \in \mathbb{R}$ yanıt değişkeni bilinmeyen bir dağılımdan çekilmiş olsun. Gaussian süreç regresyon eğitim setine ve yeni girdi verisine bağlı olarak yanıt değişkeninin değerinin tahmin edilmesi amacıyla kullanılır. Doğrusal regresyon modeli aşağıdaki şekildedir (Rasmussen ve Williams, 2006):

$$y = \mathbf{x}^\top \boldsymbol{\beta} + \varepsilon \quad (6.12)$$

Burada $\varepsilon \sim N(0, \sigma^2)$ 'dir. Hata varyansı σ^2 ve $\boldsymbol{\beta}$ katsayıları veriden tahmin edilmektedir. Gaussian süreç regresyon modeli ile yanıt değişkeninin açıklanmasında Gaussian süreç gizli değişkenler, $f(x_i), i = 1, 2, \dots, N$, ve temel fonksiyonlar, h , kullanılır. Gizli değişkenlerin kovaryans fonksiyonu yanıt değişkeninin düzgünlüğünü ifade eder ve temel fonksiyonlar x girdilerini p -boyutlu özellik uzayına yansıtır.

Gaussian süreç rasgele değişkenler setidir ki sonlu sayıda herhangi bir kümesi ortak Gaussian dağılımına sahiptir. Eğer $\{f(x), x \in \mathbb{R}^D\}$ bir Gaussian süreç ise, x_1, x_2, \dots, x_N şeklinde N gözlem verildiğinde $f(x_1), f(x_2), \dots, f(x_N)$ rasgele değişkenlerinin ortak dağılımı Gaussian olur. Gaussian süreç ortalama fonksiyonu $m(x)$ ve kovaryans fonksiyonu $k(x, x')$ ile tanımlanır. Yani eğer $\{f(x), x \in \mathbb{R}^D\}$ bir Gaussian süreç ise,

$$E(f(x)) = m(x) \quad (6.13)$$

$$\text{Cov}[f(x), f(x')] = E[\{f(x) - m(x)\}\{f(x') - m(x')\}] = k(x, x') \quad (6.14)$$

olur (Rasmussen ve Williams, 2006). Aşağıdaki model ele alınsın:

$$h(x)^\top \boldsymbol{\beta} + f(x) \quad (6.15)$$

Burada $f(x) \sim GP(0, k(x, x'))$ 'dir, yani $f(x)$, 0 ortalama ve $k(x, x')$ kovaryans fonksiyonuna sahip bir Gaussian süreçtir. $h(x)$, \mathbb{R}^D 'de tanımlı orijinal özellik vektörü x 'i, \mathbb{R}^p 'de tanımlı yeni özellik vektörü $h(x)$ 'e dönüştüren temel fonksiyonlar kümesidir. $\boldsymbol{\beta}$, $p \times 1$ boyutlu temel fonksiyon katsayılar vektörüdür. Bu model Gaussian süreç regresyon

modelini temsil etmektedir. Bir y yanıt değişkeni örneği aşağıdaki şekilde modellenebilir (Rasmussen ve Williams, 2006):

$$P(y_i|f(x_i), x_i) \sim N(y_i|h(x_i)^T \beta + f(x_i), \sigma^2) \quad (6.16)$$

Dolayısıyla Gaussian süreç regresyon olasılıksal bir modeldir. Her bir x_i gözlemi için tanımlanmış bir $f(x)$ gizli değişkeni vardır, bu da Gaussian süreç regresyon modelini nonparametrik yapmaktadır. Modelin vektör formu aşağıdaki şekildedir:

$$P(y|f, X) \sim N(y|H\beta + f, \sigma^2 I) \quad (6.17)$$

$$X = \begin{pmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_N^T \end{pmatrix}, \quad y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}, \quad H = \begin{pmatrix} h(x_1^T) \\ h(x_2^T) \\ \vdots \\ h(x_N^T) \end{pmatrix}, \quad f = \begin{pmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_N) \end{pmatrix} \quad (6.18)$$

Gaussian süreç regresyon modelindeki $f(x_1), f(x_2), \dots, f(x_N)$ gizli değişkenlerinin ortak dağılımı aşağıdaki şekildedir:

$$P = (y|f, X) \sim N(f|0, K(X, X)) \quad (6.19)$$

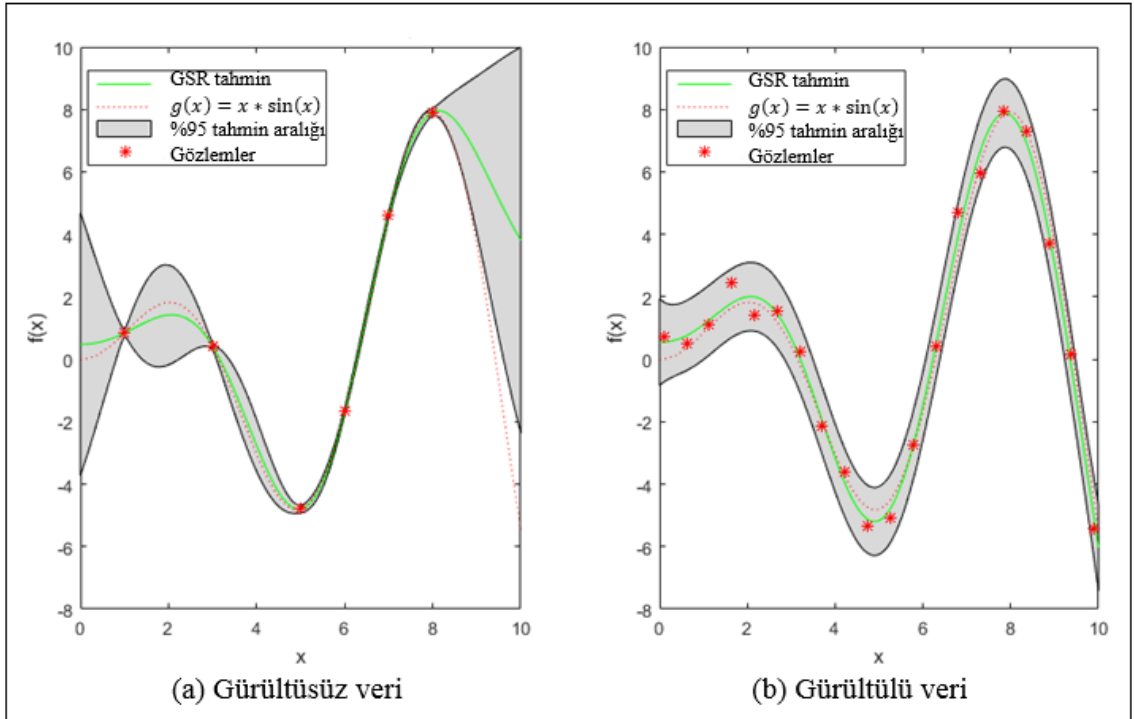
Burada $K(X, X)$ aşağıdaki şekilde ifade edilmektedir:

$$K(X, X) = \begin{pmatrix} k(x_1, x_1) & k(x_1, x_2) & \cdots & k(x_1, x_N) \\ k(x_2, x_1) & k(x_2, x_2) & \cdots & k(x_2, x_N) \\ \vdots & \vdots & \vdots & \vdots \\ k(x_N, x_1) & k(x_N, x_2) & \cdots & k(x_N, x_N) \end{pmatrix} \quad (6.20)$$

Kovaryans fonksiyonu $k(x, x')$ genellikle θ çekirdek parametreleri ya da hiperparametreler ile ifade edilir ve θ parametrelerine bağımlılığın açık olarak belirtilmesi maksadıyla $k(x, x'|\theta)$ şeklinde yazılır (Rasmussen ve Williams, 2006).

Gaussian süreç regresyon modelinin eğitilmesi ile veriden β temel fonksiyon katsayıları, σ^2 varyansı ve çekirdek fonksiyonunun θ hiperparametreleri tahmin edilir. Temel fonksiyon, çekirdek (kovaryans) fonksiyonu ve başlangıç parametre değerleri kullanıcı tarafından belirlenir.

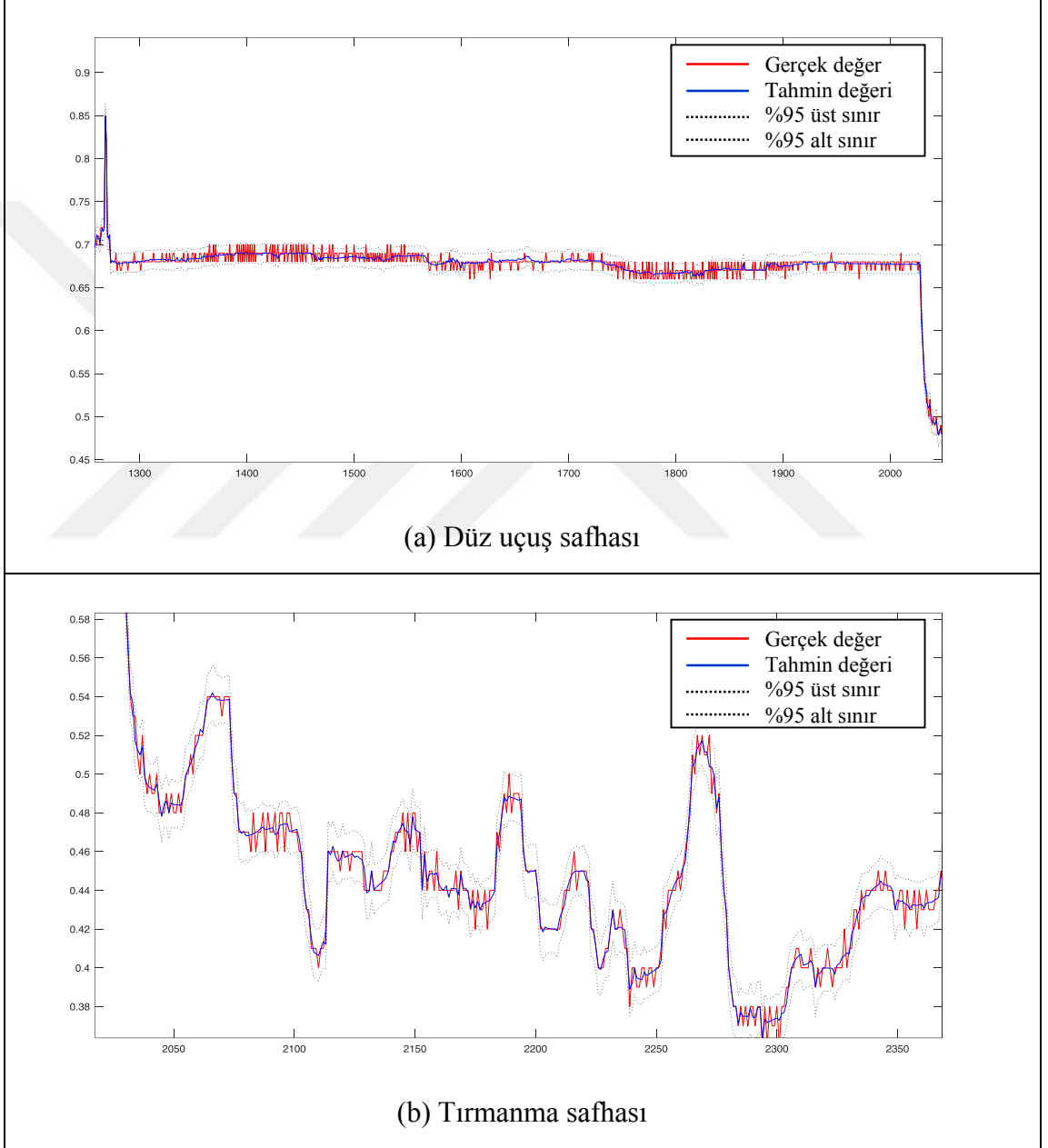
Gaussian süreç regresyon olasılıksal bir model olduğundan eğitilmiş modeli kullanarak tahmin aralıklarının hesaplanmasına olanak tanımaktadır. $g(x) = x * \sin(x)$ fonksiyonundan gözlemlenen verinin mevcut olduğu durumda elde edilen verinin gürültü içermediği varsayalım. Şekil 6.27 (a)'da bu gözlemler, Gaussian süreç regresyon ile elde edilen model ve gerçek fonksiyon değerleri gösterilmektedir. Gözlemlenen değerlerin gerçek fonksiyon değerleri değil de bunların gürültülü versiyonları olması daha gerçekçi olacaktır. Şekil 6.27 (b) bu durumu göstermektedir. Gözlemlerin gürültü içermediği durumda Gaussian süreç regresyon modeli gözlenen değerlerin üzerinden geçmekte ve tahmin edilen yanıt değişkeninin standart sapması sıfıra eşit olmaktadır. Bu sebeple söz konusu gözlem değerleri için tahmin aralığı bulunmamaktadır. Eğitilen Gaussian süreç regresyon modeli kullanılarak regresyon hata değeri de hesaplanabilmektedir.



Şekil 6.27. Gaussian süreç regresyon uyumu (MATLAB Gaussian Process Regression)

Kurulan Gaussian süreç regresyon modelinin yakıt tüketimi verisi üzerinde uygulanması ile elde edilen sonuçlar Şekil 6.28'de sunulmuştur. Görünürlüğün artırılması amacıyla grafik büyütülmüş ve yakıt tüketimindeki değişikliğin nispeten daha küçük miktarlarda gerçekleştiği düz uçuş safhası Şekil 6.28 (a)'da, yakıt tüketiminde artış ve azalışların daha büyük miktarlarda gerçekleştiği alçalış safhası Şekil 6.28 (b)'de örnek olarak

gösterilmiştir. Şekil 6.28’de gerçekleşen yakıt tüketimi kırmızı renk ile, tahmin edilen yakıt tüketim değerleri mavi renk ile gösterilmiştir. Siyah kesikli çizgiler ise %95 güven aralığını göstermektedir. Bu özellik aşırı yakıt tüketiminin belirlenmesi için etkili bir araç sunmaktadır.



Şekil 6.28. Gaussian süreç regresyon modeli ile elde edilen sonuçlar

7. SONUÇ VE ÖNERİLER

Makine öğrenme algoritmaları son zamanlarda özellikle on binlerce değişken ve/veya gözlemin mevcut olduğu veri setleri üzerinde çalışma yapan araştırmacıların ilgi odağı olmuştur. Destek vektör makineleri sınıflandırma, regresyon ve anomali tespiti problemlerinin çözümünde maliyet etkin ve yüksek performanslı sonuçlar verdiği kanıtlanmış çekirdek-tabanlı bir makine öğrenme yöntemidir. En önemli özelliklerinden birisi genel veri türleri için uygulanabilmesidir. Çekirdek fonksiyonu örnekler arasındaki iç çarpımları hesaplamaktadır. Yani doğrusal algoritmadaki iç çarpımlar çekirdek fonksiyonu ile değiştirilebilir. Böylece hem yöntemin nonlinear ilişkilerin tespitinde ve vektör olmayan veri tipleri üzerinde kullanılması sağlanmakta hem de hesaplama etkinliği kazanılmaktadır. Destek vektör makineleri maksimum marjin prensibine dayanmaktadır. Temelini istatistiksel öğrenme teorisinden almakta ve deneysel risk minimizasyonu yerine yapısal risk minimizasyonu prensibini kullanmaktadır. Bu özelliği sayesinde yerel minimuma takılma ve aşırı uyum gibi problemler aşılmıştır.

Bu çalışmada, çoklu çekirdek destek vektör regresyon yönteminin uçuş süresince anlık yakıt tüketiminin tahmini üzerinde uygulaması yapılmıştır. Yakıt tüketiminin Uçuş Veri İzleme sistemlerinde kaydedilen verilerin nonlinear bir fonksiyonu olarak modellenebileceği gösterilmiştir. İlk aşamada modelin tahmin performansını etkileyen ceza katsayısı ve hata tüpü genişliği parametrelerinin optimum değerinin belirlenmesi üzerinde durulmuştur. Elde edilen optimum parametre değerleri ile Gaussian ve Polinomial çekirdek fonksiyonları kullanılarak elde edilen modellerin yakıt tüketimini tahmin etmede başarılı sonuçlar verdiği gösterilmiştir. Özellikle düz uçuş safhasında uçuş yüksekliği ve uçağın hızının yakıt tüketimi üzerinde en etkili olan iki değişken olduğu tespit edilmiştir. Bununla birlikte bazı uçuşlarda rüzgârın yönünün yakıt tüketiminde artışa sebep olduğu gösterilmiştir.

Aşırı uyumun tespit edilmesi maksadıyla iç hatlar uçuş verileri kullanılarak kurulan model dış hatlar uçuşları üzerinde test edilmiş ve tahmin modelinin rotaya spesifik olmadığı gösterilmiştir.

Destek vektör makinelerinde en önemli problem sahası model seçimi, yani çekirdek fonksiyonunun ve optimum parametre değerlerinin seçimidir. Bu problemin aşılması için çoklu çekirdek öğrenme algoritmaları uygulanmaktadır. Çoklu çekirdek öğrenme algoritmalarında kullanıcı tarafından tek bir çekirdek fonksiyonu belirlenmesi yerine bir çekirdek fonksiyonu seti belirlenmekte ve model bu çekirdek fonksiyonlarının bir kombinasyonunu kullanmaktadır. Farklı genişlik değerlerine sahip Gaussian ve farklı dereceden Polinomial çekirdek fonksiyonlarının birleştirilmesinde pozitif katsayılar sayesinde farklı çekirdek fonksiyonlarının göreceli önem dereceleri hakkında yorum yapılabilmesine olanak tanınması sebebiyle konik kombinasyon yöntemi uygulanmıştır. Fonksiyon ağırlıkları modellerin ortalama hata kareleri ve açıklama katsayılarına bağlı olacak şekilde hesaplanmıştır. Daha sonra Gaussian ve Polinomial çekirdek fonksiyonları yaklaşık olarak aynı performansı gösterdiklerinden iki çekirdek fonksiyonunun birleştirilmesinin faydalı olacağı değerlendirilmiştir. İki fonksiyonun birleştirilmesinde ağırlıklı ortalamaları birbirini tamamlayacak ve bire eşit olacak şekilde konveks kombinasyon yaklaşımı kullanılmıştır.

Regresyon problemlerinin çözümünde kullanılan diğer makine öğrenme modellerinin aynı veri seti üzerinde uygulamaları sunulmuş, yöntemlerin birbirlerine olan üstünlükleri incelenmiştir.

Regresyon ağaçları modelinin en önemli avantajı kolay yorumlanabilir olmasıdır. Ağaç modeli değişken seçimine olanak sağlamaktadır ve veri ölçeklendirmeye ihtiyaç duyulmaz. Ancak karar ağaçları ile kurulan modelin uçak yakıt tüketiminin tahmininde doğruluk derecesinin destek vektör makinelerine kıyasla daha düşük olduğu görülmüştür. Regresyon ağaç toplulukları (bootstrap örnekleme) yönteminin uygulanması ile tek regresyon ağacı ile oluşturulan modele göre tahmin performansında iyileşme olduğu görülmektedir.

Destek vektör makinelerinin yapay sinir ağlarına en büyük üstünlüğü, yapay sinir ağları modelinde yerel minimuma takılma problemi ile karşılaşılırken, destek vektör makineleri ile her zaman global çözümün bulunmasıdır. Bununla birlikte destek vektör makineleri modeli geometrik bir yorumlamaya sahiptir ve seyrek (sparse) bir çözüm sunmaktadır. Başka bir deyişle yapay sinir ağlarının aksine, destek vektör makinelerinde modelin çözümü girdi uzayının boyutuna bağımlı olmamaktadır. Yapay sinir ağları deneysel risk minimizasyonunu kullanırken, destek vektör makinelerinde yapısal risk minimizasyonu kullanılmaktadır. Uygulamada destek vektör makineleri ile daha başarılı sonuçlar elde edilmesinin sebebinin yapay sinir ağlarının en büyük problemi olan aşırı uyum sorununun aşılmasında daha etkin çözüm sunması olduğu değerlendirilmektedir.

Gaussian süreç regresyon modelinin en büyük avantajı tahmin için güven aralığı değerlerinin de hesaplanmasına olanak sağlamasıdır. Bu özellik anomali tespiti, mevcut problemde aşırı yakıt tüketiminin tespiti, konusunda önem arz etmektedir. Ancak kovaryans matrisinin tersinin hesaplanmasını gerektirdiğinden özellikle büyük boyutlu veri durumunda hafıza yetersizliği gibi problemlere sebep olmakta ve zaman açısından maliyetli olmaktadır. Elde edilen sonuçlar Çizelge 7.1’de sunulmuştur.

Çizelge 7.1. Regresyon model sonuçlarının karşılaştırılması

Regresyon modeli	HKO	NHKOK
Destek vektör regresyon	0.0001	0.0532
Regresyon ağaçları	0.0019	0.0785
Regresyon ağaç toplulukları (bootstrap örnekleme)	0.0014	0.0689
Yapay sinir ağları	0.0007	0.0635
Gaussian süreç regresyon	0.0010	0.0567

Sonuç olarak, makine öğrenme regresyon yöntemlerinin uçaklarda anlık yakıt tüketiminin tahmini probleminin çözümünde başarılı bir şekilde uygulanabileceği gösterilmiştir. Destek vektör regresyon ile kurulan model en yüksek performans gösteren sonuçları vermiştir. Yakıt tüketimine ilişkin açıklayıcı değişkenlerin her saniye kaydedildiği ve parametre sayısının fazlalığı göz önünde bulundurulduğunda destek vektör makineleri yönteminin seyrek (sparse) sonuçlar vermesi ve modeli verinin boyutundan bağımsız olarak sadece destek vektör olarak seçilen gözlemlere dayanarak oluşturması maliyet etkin çözümler sunmaktadır. Destek vektör makinelerinin en önemli dezavantajı ise değişken seçimi ve değişkenlerin önem derecelerinin belirlenmesi konusundadır. Bu problemin çözümünde karar ağaçları ve lasso yöntemlerinden faydalanılmıştır. Bununla birlikte Gaussian süreç regresyon modelinin uygulanması ile elde edilen tahmin için güven aralıkları anomali tespiti kapsamında faydalı bir araç sunmaktadır.

Nonlinear çekirdek tabanlı destek vektör makinelerinin kullanılması uçuşun tüm safhalarının istatistiksel analizini mümkün kılmıştır. Bu yöntem mevcut durumda uygulanan sadece toplam yakıt tüketim miktarının planlanan toplam yakıt tüketim miktarı ile karşılaştırılmasından ziyade uçuş süresince tüketilen yakıt miktarının modellenmesi ile gerçekleşen ve beklenen anlık yakıt tüketim miktarlarının analizine dayanmaktadır. Elde edilen regresyon modelinden faydalanılarak, gerçekleşen yakıt tüketiminin beklenen değerin üzerinde olduğu durumlar ve aşırı yakıt tüketimine neden olan faktörler tespit edilebilmektedir. Uygulanan yöntem incelenmek istenen nicel bir değişkenin ve bu değişken üzerinde etkisi olan kategorik veya nicel değişkenlerin zaman serisi ölçümlerinin mevcut olduğu tüm sistemler için kullanılabilir.

KAYNAKLAR

- Ayhan, S.**, 2013. Kaba Küme ve Destek Vektör Makineleri Kullanılarak Nitelik İndirgeme ve Sınıflandırma Problemlerinin Çözümü için Bütünleşik Bir Yaklaşım, *Doktora Tezi*, Osmangazi Üniversitesi Fen Bilimleri Enstitüsü, Eskişehir.
- Bach, F. R., Lanckriet, G. R. G. and Jordan M. I.**, 2004. Multiple Kernel Learning, Conic Duality, and the SMO Algorithm, *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada.
- Bi, J., Bennett, K., Embrechts, M., Breneman, C. and Song, M.**, 2003. Dimensionality reduction via sparse support vector machines, *JMLR*, **3**, 1229-1243.
- Blanz, V., Schölkopf, B., Bühlhoff, H., Burges, C., Vapnik, V. and Vetter, T.**, 1996. Comparison of view-based object recognition algorithms using realistic 3D models, *Artificial Neural Networks - ICANN'96*, Germany, 251-256.
- Boeing - AERO Magazine**, What is Angle of Attack, 13 Aralık 2015.
http://www.boeing.com/commercial/aeromagazine/aero_12/whatisaoa.pdf.
- Boser, B. E., Guyon, I. M. and Vapnik, V. N.**, 1992. A Training Algorithm for Optimal Margin Classifiers, *Proceedings of the Annual Conference on Computational Learning Theory*, Pittsburgh, PA, USA, 144-152.
- Breiman, L., Friedman, J., Olshen, R., and Stone, C.**, 1984. Classification and Regression Trees, CRC Press, Boca Raton, FL.
- Breiman L.**, 1996. Bagging predictors. *Machine Learning*, **24(2)**, 123-140
- Burges, C. J. C.**, 1998. A Tutorial on Support Vector Machines for Pattern Recognition, *Data Mining and Knowledge Discovery*, **2**, 121-167.
- Burges, C. J. C. and Schölkopf, B.**, 1997. Improving the accuracy and speed of support vector learning machines, *Advances in Neural Information Processing Systems*, **9**, 375-381.

- Chang, C.C. and Lin, C.J.**, 2002. Training ν -Support Vector Regression: Theory and algorithms. *Neural Computation*, **14**, 1959-1977.
- Chang, C.-C. and Lin, C.-J.**, 2011. LIBSVM: A Library for Support Vector Machines, *ACM Transactions on Intelligent Systems and Technology*. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- Cortes, C. and Vapnik, V.**, 1995. Support-Vector Networks, *Machine Learning*, **20**, 273-297.
- Cristianini, N. and Shawe-Taylor, J.**, 2000, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, Cambridge, UK.
- Cristianini, N. and Shawe-Taylor, J.**, 2004. Kernel Methods for Pattern Analysis, Cambridge University Press, Cambridge, UK.
- Das, K., Bhaduri K. and Oza, N.**, 2014. ParitoSVR: Parallel Iterated Optimizer for Support Vector Regression in the Primal, *2014 SDM Workshop on Optimization Methods for Anomaly Detection*, Prague, Czech Republic.
- Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A. and Vapnik, V.**, 1997. Support vector regression machines, *Advances in Neural Information Processing Systems*, **9**, 155-161.
- Fletcher, R.**, 2000. Practical Methods of Optimization. 2nd edition. Wiley, New York.
- Flight Controls**, Trailing Edge Flaps, 10 Temmuz 2015.
http://www.b737.org.uk/flightcontrols.htm#Trailing_Edge_Flaps.
- Friedman, J., Tibshirani, R., and Hastie, T.**, 2010. Regularization Paths for Generalized Linear Models via Coordinate Descent, *Journal of Statistical Software*, **33(1)**.
- Gönen, M. and Alpaydın, E.**, 2008. Localized Multiple Kernel Learning, *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland.
- Gönen, M.**, 2010. Localized Multiple Kernel Learning Algorithms for Machine Learning, *Doktora Tezi*, Boğaziçi Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Gönen, M. and Alpaydın, E.**, 2010. Localized Multiple Kernel Regression, *Proceedings of the International Conference on Pattern Recognition*.

- Gönen, M. ve Alpaydın, E.**, 2011. Multiple Kernel Learning Algorithms, *Journal of Machine Learning Research*, **12**, 2211-2268.
- Guajardo, J. A., Weber, R. and Miranda, J.**, 2006. A Forecasting Methodology Using Support Vector Regression and Dynamic Feature Selection, *Journal of Information & Knowledge Management*, **5(4)**, 329-335.
- Guajardo, J. A., Weber, R. and Miranda, J.**, 2010. A model updating strategy for predicting time series with seasonal patterns, *Applied Soft Computing*, **10**, 276-283.
- Gunn, S.R.**, 1998. Support Vector Machines for Classification and Regression, *Technical Report*, University of Southampton.
- Guyon, I., Weston, J., Barnhill, S. and Vapnik, V.**, 2002. Gene selection for cancer classification using support vector machines. *Machine Learning*, **46(1-3)**, 389-422.
- Guyon, I. and Elisseeff, A.**, 2003. An Introduction to Variable and Feature Selection, *Journal of Machine Learning Research*, **3**, 1157-1182.
- Huber, P. J.**, 1964. Robust Estimation of a Location Parameter, *Ann. Math. Statist.*, **35(1)**, 73-101.
- Hsu, C.-W. and Lin, C.-J.**, 2002. A Comparison of Methods for Multiclass Support Vector Machines, *IEEE Transactions on Neural Networks*, **13(2)**, 415-425.
- Hsu, C.-W., Chang, C.-C. and Lin, C.-J.**, 2010. A Practical Guide to Support Vector Classification, Department of Computer Science, National Taiwan University.
- ICAO Carbon Emissions Calculator Methodology**, 2015. 15 Aralık 2015.
<http://www.icao.int/ENVIRONMENTAL-PROTECTION/CARBONOFFSET/Pages/default.aspx>.
- ICAO Standards and Recommended Practices (SARPs)**, Annex 6, 13 Aralık 2014.
<http://www.icao.int/safety/ism/ICAO%20Annexes/Annex%206.pdf>.
- ICAO Working Paper**, Improvement of Flight Data Monitoring Systems in Turkey, 2013. http://www.icao.int/Meetings/a38/Documents/WP/wp264_en.pdf.
- Joachims, T.**, 1997. Text categorization with support vector machines, *Technical Report, LS VIII Number 23*, University of Dortmund.

- Joachims, T., Cristianini, N. and Shawe-Taylor J.**, 2001. Composite Kernels for Hypertext Categorisation, *Proceedings of the 18th International Conference on Machine Learning*.
- Kabaoğlu, R. O.**, 2010. Destek Vektörü Makineleri Tabanlı Hata Bulma, Tanıma ve Hata Toleranslı Kontrol Yöntemleri, *Doktora Tezi*, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Karagiannopoulos, M., Anyfantis, D., Kotsiantis, S. B. and Pintelas, P. E.**, 2007. Feature Selection for Regression Problems. *Proceedings of HERCMA'07*.
- Karal, Ö.**, 2011. Gürbüz ve Yalın Regresyon için Parça Parça Doğrusal ve Destek Vektör Tabanlı Modeller, *Doktora Tezi*, Dokuz Eylül Üniversitesi Fen Bilimleri Enstitüsü, İzmir.
- Kim, E.**, 2013. The Kernel Trick, 13 Aralık 2014.
http://www.eric-kim.net/eric-kim-net/posts/1/kernel_trick.html.
- Kohavi, R. and John, G.**, 1997. Wrappers for feature selection. *Artificial Intelligence*, **1(2)**, 273–324.
- Lanckriet, G. R. G., Cristianini, N., Bartlett, P., Ghaoui, L. E. and Jordan, M. I.**, 2004. Learning the kernel matrix with semidefinite programming, *J. Machine Learning Research*, **5**, 27-72.
- LeCun, Y., Denker, J., Solla, S., Howard, R. E. and Jackel, L. D.**, 1989. Optimal brain damage, *Proceedings of the Neural Information Processing Systems*.
- Lewis, D. P., Jebara, T. and Noble, W. S.**, 2006. Support Vector Machine Learning from Heterogeneous Data: An Empirical Analysis Using Protein Sequence and Structure, *Bioinformatics*, **22**, 2753-2760.
- Maldonado, S. and Weber, R.**, 2010. Feature Selection for Support Vector Regression via Kernel Penalization, *Proceedings of the 2010 International Joint Conference on Neural Networks*, Barcelona, Spain, 1973-1979.
- Matlab Statistics Toolbox**, 2016. The Mathworks.
- Mercer J.**, 1909. Functions of positive and negative type and their connection with the theory of integral equations, *Philosophical Transactions of the Royal Society*, **209**, 415-446.
- Mitchell, T.**, 1997. Machine Learning, McGraw Hill.

- Mukherjee, S., Osuna, E. and Girosi, F.,** 1997. Nonlinear prediction of chaotic time series using a support vector machine, *In Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, Amelia Island, FL, 511-519.
- Müller, K.R., Smola, A., Ratsch, G., Schölkopf, B., Kohlmorgen, J. and Vapnik, V.,** 1997. Predicting time series with support vector machines, *In Proceedings of the International Conference on Artificial Neural Networks*.
- Nabney, I.T.,** 2001. NETLAB-algorithms for pattern recognition, Springer, New York
- NextGen,** 2013. Phases of Flight, Federal Aviation Administration.
<http://www.faa.gov/nextgen/library/images/phasesOfFlight2013.jpg>.
- Osuna, E., Freund, R. and Girosi, F.,** 1997a. Training support vector machines: an application to face detection, *In IEEE Conference on Computer Vision and Pattern Recognition*, 130-136.
- Osuna, E., Freund, R., and Girosi, F.** 1997b. An improved training algorithm for support vector machines. *Neural Networks for Signal Processing*, 276-285.
- Oza, N. C. and Das K.,** 2014. Reducing the Environmental Impact of Aviation: A Data Mining Approach to Instantaneous Estimation of Fuel Consumption, NASA Ames Research Center.
- Perkins, S., Lacker, K. and Theiler, J.,** 2003. Grafting: Fast incremental feature selection by gradient descent in function space, *JMLR*, **3**, 1333-1356.
- Platt, J.,** 1999. Fast training of support vector machines using sequential minimal optimization, in *Advances in Kernel Methods - Support Vector Learning*, 185–208, Eds. Schölkopf B., Burges, C.J.C. and Smola, A.J., MIT Press, Cambridge, MA.
- Qiu, S. and Lane, T.,** 2005. Multiple kernel learning for support vector regression, *Technical Report*, Computer Science Department, University of New Mexico.
- Qiu, S. and T. Lane,** 2009. A Framework for Multiple Kernel Support Vector Regression and Its Applications to siRNA Efficacy Prediction, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **6(2)**, 190-199.

- Rakotomamonjy, A.**, 2003. Variable selection using SVM-based criteria, *JMLR*, **3**, 1357-1370.
- Rakotomamonjy, A., Bach, F., Canu, S. and Grandvalet, Y.**, 2007. More Efficiency in Multiple Kernel Learning, *In Proceedings of the 24th International Conference on Machine Learning*.
- Rakotomamonjy, A., Bach, F. R., Canu, S. and Grandvalet, Y.**, 2008. SimpleMKL, *Journal of Machine Learning Research*, **9**, 2491-2521.
- Rasmussen, C. E. and Williams, C. K. I.**, 2006. Gaussian Processes for Machine Learning, MIT Press, Cambridge, Massachusetts.
- Rivals, I. and Personnaz, L.**, 2003. MLPs (mono-layer polynomials and multi-layer perceptrons) for non-linear modeling, *JMLR*, **3**, 1383-1398.
- Samuel, A.**, 1959. Some Studies in Machine Learning Using the Game of Checkers, *IBM Journal*, **3(3)**, 210-229.
- Schmidt, M.**, 1996. Identifying speaker with support vector networks, *In Proceedings of Interface '96*, Sydney.
- Schölkopf, B., Burges, C. and Vapnik, V.**, 1995. Extracting support data for a given task, *In Proceedings of First International Conference on Knowledge Discovery & Data Mining*, Menlo Park, CA.
- Schölkopf, B., Burges, C. and Vapnik, V.**, 1996. Incorporating invariances in support vector learning machines, *Artificial Neural Networks - ICANN'96*, **1112**, 47-52, Berlin.
- Schölkopf, B., Smola A., Williamson, R.C. and Bartlett, P.L.**, 2000. New support vector algorithms, *Neural Computation*, **12**, 1207-1245.
- Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J. and Williamson, R. C.**, 2001. Estimating the support of a high-dimensional distribution. *Neural Computation*, **13(7)**, 1443-1471.
- Schölkopf, B. and Smola, A.J.**, 2002. Learning with Kernels, MIT Press.
- Smola, A. J. and Schölkopf, B.**, 2004. A Tutorial on Support Vector Regression, *Statistics and Computing*, **14**, 199-222.

- Sonnenburg, S., Ratsch, G. and Schafer, C.**, 2006a. A General and Efficient Multiple Kernel Learning Algorithm, *Advances in Neural Information Processing Systems*, **18**.
- Sonnenburg, S., Ratsch, G., Schafer, C. and Scholkopf, B.**, 2006b. Large Scale Multiple Kernel Learning, *Journal of Machine Learning Research*, **7**, 1531-1565.
- Sonnenburg, S., Ratsch, G., Henschel, S., Widmer, C., Behr, J., Zien, A., Bona, F. de, Binder, A., Gehl, C. and Franc, V.**, 2010. The SHOGUN Machine Learning Toolbox. *Journal of Machine Learning Research*, **11**, 1799-1802.
- Srivastava, A. N.**, 2012. Greener Aviation with Virtual Sensors: A Case Study, *Data Mining and Knowledge Discovery*, **24(2)**, 443-471.
- Teledyn Controls**, Flight Data Monitoring Solutions, Flight Data Monitoring Guide. <http://www.teledynecontrols.com/pdf/FDMS-Brochure.pdf>. 13 Aralık 2014.
- Tibshirani, R.**, 1996. Regression Shrinkage and Selection via the Lasso, *Journal of the Royal Statistical Society*, **58(1)**, 267-288.
- Vanderbei R.J.**, 1998. LOQO: An interior point code for quadratic programming, *Technical Report*, Statistics and Operations Research, Princeton University.
- Varma, M. and Babu, B. R.**, 2009. More Generality in Efficient Multiple Kernel Learning, *In Proceedings of the 26th International Conference on Machine Learning*.
- Vapnik, V.N.**, 1982. Estimation of Dependences Based on Empirical Data. Springer, Berlin.
- Vapnik, V.**, 1995. The Nature of Statistical Learning Theory, Springer-Verlag, New York.
- Vapnik, V.**, 1998. Statistical Learning Theory, John Wiley and Sons, New York.
- Yang, J.B. and Ong, C.J.**, 2010. Feature Selection for Support Vector Regression Using Probabilistic Prediction, *KDD'10*, Washington, DC, USA.
- Weston, J., Elisseeff, A., Schölkopf, B. and Tipping M.**, 2003. Use of the zero norm with linear models and kernel methods, *JMLR*, **3**, 1439-1461.

Flap (aeronautics), In Wikipedia, [https://en.wikipedia.org/wiki/Flap_\(aeronautics\)](https://en.wikipedia.org/wiki/Flap_(aeronautics)).
10 Temmuz 2015.

Woodbury, T. and Srivastava, A. N., 2012. Analysis of Virtual Sensors for Predicting Aircraft Fuel Consumption, American Institute of Aeronautics and Astronautics.

Zhao, H. X. and Magoules, F., 2011. Feature selection for support vector regression in the application of building energy prediction, *9th IEEE International Symposium on Applied Machine Intelligence and Informatics (SAMII 2011)*, January 2011, Smolenice, Slovakia.



Cansın YALÇIN

EĞİTİM DURUMU

Mimar Sinan Üniversitesi Fen Bilimleri Enstitüsü <i>Doktora - İstatistik Bölümü</i>	2012-2016
University of Southern California - Los Angeles, CA, USA <i>Fulbright Bursiyeri - Doktora Tez Araştırma</i>	2014-2015
Mimar Sinan Üniversitesi Fen Bilimleri Enstitüsü <i>Yüksek Lisans - İstatistik Bölümü</i>	2010-2012
Kara Harp Okulu <i>Lisans - Sistem (Endüstri) Mühendisliği</i>	2005-2009

İŞ DENEYİMİ

NATO Hızlı Konuşlanabilir Kolordu (Rapid Deployable Corps) <i>Toplumsal Cinsiyet Danışmanı (Gender Advisor)</i>	2014-Hâlen
3'üncü Kolordu Komutanlığı <i>Taşınır Dayanıklı Mal Saymanı</i>	2011-2014
MSB İstanbul Tedarik Bölge Başkanlığı <i>Tedarik Plan ve Koordinasyon Subayı</i>	2010-2011

YABANCI DİL: İngilizce (YDS 96.25 - TOEFL iBT 102)

İletişim: yalcin.cansin@outlook.com